

**Βιβλίο Λύσεων**

**Μαθητικός  
Διαγωνισμός  
Πληροφορικής  
και Υπολογιστικής Σκέψης**

***Κάστορας –  
Bebras* ®**

**2024 - 2025**



## Ευχαριστίες

Θέλουμε να ευχαριστήσουμε τους 27.000 μαθητές που έλαβαν μέρος στον διαγωνισμό Bebras 2024-2025. Εκφράζουμε τις θερμές ευχαριστίες μας στην Διεθνή Κοινότητα Bebras, που επέτρεψε να χρησιμοποιήσουμε τα θέματα του φετινού διαγωνισμού 2024-2025.

Ευχαριστούμε ιδιαίτερος τους συνδιοργανωτές μας, τη 'Μονάδα Αρχιμήδης', το Ινστιτούτο Εκπαιδευτικής Πολιτικής (ΙΕΠ), την openAIRE και την ΓΓΕΚ για την παραχώρηση της αιγίδας της.

Για την υποστήριξη και προώθηση του Διαγωνισμού στην Ελλάδα, ευχαριστούμε πολύ τους υποστηρικτές φορείς DARELAB, την ΕΠΥ, το LTEE lab του Πανεπιστημίου Αιγαίου, την ΕΕΛ/ΛΑΚ και τα μέλη της επιστημονικής επιτροπής του Διαγωνισμού.

Ευχαριστούμε επίσης τα μέλη της οργανωτικής επιτροπής για την αδιάκοπη συνεισφορά τους, Φεσάκη Γεώργιο, Βουρλέτση Ιωάννη, Δουκάκη Σπυρίδων, Ζέρβα Κωνσταντίνο, Καραλή Ολυμπία, Κολέζα Ευγενία, Νικολαΐδη, Παπαδάκη Σταμάτη, Παπασπύρου Νίκο, Πραντσούδη Σταυρούλα, Σαλπασαράνη Κωνσταντίνο, Τζάμο Χρήστο, Φωτάκη Δημήτρη και Ψυχάρη Σαράντο.

Τέλος, για την συμβολή και δημιουργία του βιβλίου λύσεων ευχαριστούμε τους Καζανά Σταυρούλα, Καραγιάννη Αστέριο, Καραπά Βασίλη, Μπέη Ελένη, Μπόνταρε Φίλιππο και Σάρρα Πασγάλη.

Τέλος, εκφράζουμε τις θερμότερες των ευχαριστιών μας στους εκπαιδευτικούς-συντονιστές σχολικών μονάδων, για την προώθηση και χρήση του Διαγωνισμού στην εκπαιδευτική διαδικασία. Με μεγάλη χαρά πληροφορηθήκαμε τον ενθουσιασμό των μαθητών για τον Διαγωνισμό και ανυπομονούμε για τον επόμενο. Σας καλούμε να συνεχίσετε μαζί μας το ταξίδι στον κόσμο της Υπολογιστικής Σκέψης και σας περιμένουμε ξανά κοντά μας τον Νοέμβριο.

Με εκτίμηση,

Πετρά Ελένη

Υπεύθυνη της Μονάδας Καινοτομίας για τις Γυναίκες, Ερευνητικό Κέντρο Αθηνά

## Πίνακας Περιεχομένων

Α/Α	Κωδικός	Τίτλος	Ηλικιακή Ομάδα						Σελίδα
			6-8	8-10	10-12	12-14	14-16	16-18	
1.	2024-ID-04-GR	Ο κάστορας χάθηκε	E						6
2.	2024-LT-01-GR	Σερβίτσιο στο ράφι	E						10
3.	2024-DE-02-GR	Πάρτυ με πίτσα	E						13
4.	2024-NL-02-GR	Το δώρο των γενεθλίων	E						18
5.	2024-BR-04-GR	Οργάνωση βραχιολιών	M						21
6.	2024-IE-01c-GR	Περιστρεφόμενα τριφύλλια	M						24
7.	2024-PL-03-GR	Σχεδιάζοντας ένα ιστιοφόρο	M						27
8.	20124-KR-02-GR	Πάρτυ στην αυλή	Δ						30
9.	2024-MY-03-GR	Διαδικτυακή τάξη	Δ						34
10.	2024-SK-01b-GR	Η κουδουνίστρα του Άλκη	Δ						38
11.	2024-ID-04-GR	Ο κάστορας χάθηκε		E					41
12.	2024-IE-01c-GR	Περιστρεφόμενα τριφύλλια		E					44
13.	2024-LT-01-GR	Σερβίτσιο στο ράφι		E					47
14.	2024-PL-03-GR	Σχεδιάζοντας ένα ιστιοφόρο		E					50
15.	2024-SK-01b-GR	Η κουδουνίστρα του Άλκη		M					53
16.	2024-CH-03b-GR	Κάρτες		M					56

17.	20124-KR-02-GR	Πάρτυ στην αυλή		M					59
18.	2024-PL-04-GR	Η διαδρομή του Ρομπότ		Δ					63
19.	2024-TW-03-UK-GR	Το πιο μακρύ βραχιόλι		Δ					67
20.	2024-BG-01b-GR	Μπάλες		Δ					70
21.	2024-FI-01-GR	Σελίδα ζωγραφικής			E				75
22.	2024-KR-02-GR	Πάρτυ στην αυλή			E				79
23.	2024-KR-03a-GR	Σωλήνας			E				83
24.	2024-MT-02-GR	Δωρεάν θέσεις στάθμευσης			E				87
25.	2024-DE-03-GR	Το λουλούδι θαύμα			M				90
26.	2024-PL-04-GR	Η διαδρομή του Ρομπότ			M				93
27.	2024-BG-01b-GR	Μπάλες			M				97
28.	2024-BE-01a-GR	Τουβλάκια στην σειρά			Δ				102
29.	2024-SI-01-GR	Επιστροφή Βιβλίων			Δ				105
30.	2024-TW-03-UK-GR	Το πιο μακρύ βραχιόλι			Δ				108
31.	2024-PL-04-GR	Η διαδρομή του Ρομπότ				E			111
32.	2024-CH-03a-GR	Κάρτες Ricca				E			115
33.	2024-DE-04a-GR	Ηλιόλουστες Μέρες				E			119
34.	2024-BG-01b-GR	Μπάλες				E			122
35.	2024-DE-07-GR	Ηλεκτρονικό Παιχνίδι				M			127
36.	2024-DE-08-GR	Ξενάγηση Δέντρων				M			131
37.	2024-TW-03-UK-GR	Το πιο μακρύ βραχιόλι				M			135
38.	2024-SK-04-GR	Ακολουθία Γραμμάτων				Δ			138

39.	2024-CA-02-GR	Αλυσίδες Λέξεων				Δ			143
40.	2024-IT-01b-GR	Χάρτης Φαγητού				Δ			146
42.	2024-SI-01-GR	Επιστροφή βιβλίων					E		151
42.	2024-DE-07-GR	Ηλεκτρονικό παιχνίδι					E		154
43.	2024-HU-03	Κρυμμένος Θησαυρός					E		157
44.	2024-TW-03-UK-GR	Το πιο μακρύ βραχιόλι					E		160
45.	2024-SK-04	Ακολουθία Γραμμάτων					M		163
46.	2024-CA-02-GR	Αλυσίδες Λέξεων					M		168
47.	2024-LT-05	Φίλοι					M		171
48.	2024-PK-03-GR	Σιδηροδρομικό Δίκτυο					Δ		174
49.	2024-DE-06 a-GR	Μηχανή Μπαλονιών					Δ		177
50.	2024-IT-01b-GR	Χάρτης Φαγητού					Δ		181
51.	2024-SK-04	Ακολουθία Γραμμάτων						E	186
52.	2024-CA-02-GR	Αλυσίδες Λέξεων						E	191
53.	2024-LT-05	Φίλοι						E	194
54.	2024-IT-01b-GR	Χάρτης Φαγητού						E	197
55.	2024-AU-03 b-GR	Σφάλμα στην ζωγραφική						M	202
56.	2024-FI-03	Τοίχος από τούβλα						M	207
57.	2024-PK-03-GR	Σιδηροδρομικό Δίκτυο						M	212
58.	2024-AU-01-GR	Κρυπτογραφημένη Εικόνα						Δ	215
59.	2024-DE-06 a-GR	Μηχανή Μπαλονιών						Δ	220
60.	2024-HU-04-GR	Palago						Δ	224




## 2024-ID-04 – Ο κάστορας χάθηκε

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Ο κάστορας χάθηκε

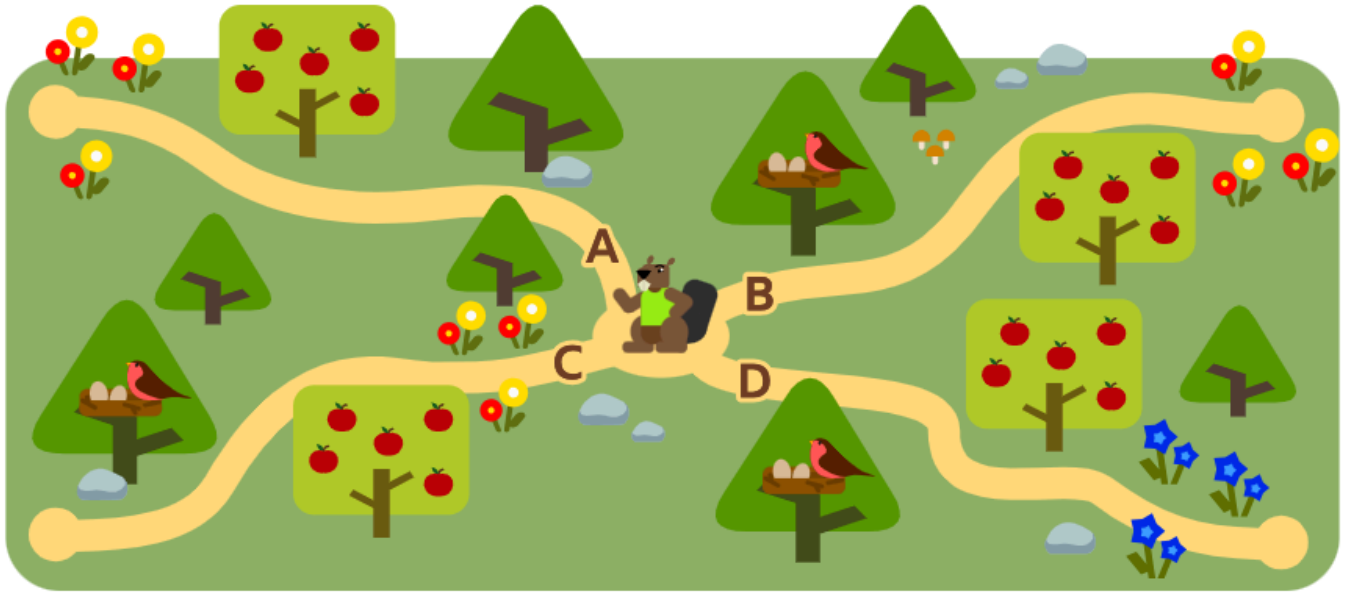
Ο Κάστορας Ελένη είναι στο δάσος. Στο δρόμο συνάντησε τα εξής:

Πρώτα, είδε κόκκινα και κίτρινα λουλούδια.	
Στην συνέχεια, είδε μια μηλιά.	
Τέλος, είδε μια φωλιά πουλιού σε ένα δέντρο.	

### Ερώτηση

Κάνε κλικ στο γράμμα στο μονοπάτι που χρησιμοποιεί η Ελένη για να μπει στο δάσος.

*Κάνε κλικ στην επιλογή "Αποθήκευση" όταν τελειώσεις.*



## Απάντηση

Η σωστή απάντηση είναι το μονοπάτι B.







## Εξήγηση απάντησης

Στο δρόμο προς το δάσος, η Ελένη βλέπει, πρώτα, κόκκινα και κίτρινα λουλούδια, έπειτα, βλέπει μια μηλιά και στο τέλος, μια φωλιά πουλιών σε ένα δέντρο. Στο δρόμο για το σπίτι, βλέπει τα ίδια αντικείμενα, αλλά με την αντίθετη σειρά. Έτσι, στο δρόμο για το σπίτι, βλέπει πρώτα μια φωλιά πουλιών, μετά μια μηλιά και τέλος τα κόκκινα και κίτρινα λουλούδια. Ως εκ τούτου, το B είναι η σωστή απάντηση.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Αυτό το πρόβλημα προσομοιώνει μια δομή δεδομένων στοίβας (stack) που χρησιμοποιεί τη διαδικασία LIFO (Last In First Out), δηλαδή "Τελευταίο που μπαίνει, πρώτο που βγαίνει". Τα αντικείμενα που βλέπει η Ελένη μπορούν να θεωρηθούν ως αντικείμενα που τοποθετούνται σε μια στοίβα. Το πρώτο αντικείμενο που είδε (κόκκινα και κίτρινα λουλούδια) τοποθετείται σε μια στοίβα, ακολουθούμενο από το δεύτερο αντικείμενο (μηλιά) και το τρίτο αντικείμενο (φωλιά πουλιού πάνω στο δέντρο), έτσι ώστε η στοίβα να περιγράφεται ως εξής:

Το πρώτο που είδε η Ελένη	Το δεύτερο που είδε η Ελένη	Το τρίτο που είδε η Ελένη
		
		
		

Όταν η Ελένη περπατά προς την αντίθετη κατεύθυνση, κάθε αντικείμενο στην στοίβα πρέπει να αφαιρεθεί. Το πρώτο πράγμα που θα δει στον δρόμο της επιστροφής η Ελένη είναι μια φωλιά πουλιού σε ένα δέντρο. Στη συνέχεια, η φωλιά αφαιρείται από την στοίβα, ώστε στην κορυφή της στοίβας να βρίσκεται η μηλιά. Έπειτα, αφαιρείται και η μηλιά από την στοίβα. Το τελευταίο αντικείμενο που αφαιρείται από την στοίβα είναι τα κόκκινα και κίτρινα λουλούδια.

#### Υπολογιστική σκέψη

Για να λύσουμε αυτό το πρόβλημα πρέπει να επικεντρωθούμε σε ορισμένα πράγματα και να αγνοήσουμε άλλα. Χρειάζεται να δώσουμε προσοχή μόνο στα κόκκινα και κίτρινα λουλούδια, τη μηλιά και τη φωλιά του πουλιού. Όλα τα υπόλοιπα δεν μας βοηθούν να λύσουμε το πρόβλημα, οπότε μπορούμε να τα παραβλέψουμε. Αυτό ονομάζεται αφαίρεση.



Χώρα προέλευσης θέματος

Ινδονησία

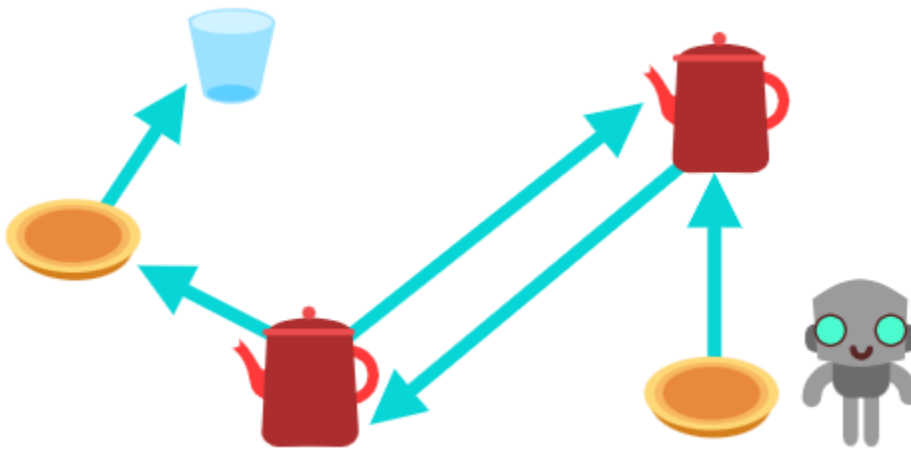
## 2024-LT-01 – Σερβίτσιο στο ράφι

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Σερβίτσιο στο ράφι

Ένα ρομπότ τοποθετεί το σερβίτσιο στα ράφια της κουζίνας. Εάν χρησιμοποιήσει τους κανόνες που φαίνονται στο παρακάτω σχήμα:



### Ερώτηση

Σε ποιο ράφι είναι το σερβίτσιο που δεν είναι στη σωστή σειρά;

Πάτησε στο γράμμα δίπλα στο σωστό ράφι. Πάτησε "Αποθήκευση" όταν τελειώσεις.



## Απάντηση

Η απάντηση είναι το γράμμα C.

## Εξήγηση απάντησης

Παρατηρήσατε ότι ο αριθμός των τσαγιερών σε μια σειρά στο ράφι πρέπει να είναι ζυγός; Αυτό συμβαίνει επειδή το διάγραμμα δείχνει επανάληψη με τις τσαγιέρες: όταν η πρώτη τσαγιέρα τοποθετείται, πρέπει να τοποθετηθεί και η δεύτερη τσαγιέρα, όταν η τρίτη τσαγιέρα τοποθετείται, πρέπει να τοποθετηθεί και η τέταρτη και ούτω καθεξής.

Μόνο η επιλογή C έχει μονό αριθμό τσαγιερών, οπότε είναι λανθασμένη. Φυσικά, χρειάζεται να ελέγξετε προσεκτικά τις επιλογές A, B και D για να βεβαιωθείτε ότι το σερβίτσιο είναι τοποθετημένο σύμφωνα με το δεδομένο διάγραμμα.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Αυτό είναι ένα κλασικό πρόβλημα της επιστήμης των υπολογιστών, όπου σας δίνεται ένα σχέδιο (σε αυτή την περίπτωση ένα λεγόμενο αυτόματο) και πρέπει να κατανοήσετε πώς λειτουργεί και ποια αποτελέσματα μπορούν να προκύψουν. Ο πιο εύκολος τρόπος για να το κάνετε αυτό είναι να κοιτάξετε μερικά παραδείγματα – τότε μπορείτε να δείτε τα πρότυπα. Έτσι, όταν ψάχνουμε για μια παραλλαγή στην οποία το σερβίσιο δεν είναι τοποθετημένο σωστά σύμφωνα με το σχέδιο, θα εστιάσουμε σε ένα από τα πιο σημαντικά μέρη του σχεδίου – την επανάληψη (τοποθέτηση των τσαγιερών). Πρέπει να διασφαλίσουμε ότι ο αριθμός των τσαγιερών είναι πάντα ζυγός στη σειρά των πιάτων.

### Υπολογιστική σκέψη

Σε αυτό το πρόβλημα, όταν ελέγχουμε την ορθότητα των επιλογών (τη σειρά των πιάτων) σύμφωνα με το δεδομένο διάγραμμα, πρέπει να παρατηρήσουμε την αναγνώριση προτύπων, που αποτελεί ένα από τα σημαντικά συστατικά της υπολογιστικής σκέψης. Η αναγνώριση προτύπων, τάσεων και κανονικοτήτων μέσα στο πρόβλημα ή στη λύση του βοηθά στην καλύτερη κατανόηση του προβλήματος και στην εύρεση πιο αποδοτικών λύσεων.

## Χώρα προέλευσης θέματος

Λιθουανία

## 2024-DE-02 – Πάρτυ με πίτσα

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Πάρτυ με πίτσα

Ο Γιάννης διοργανώνει ένα πάρτυ με πίτσα για τους φίλους του. Ξέρει τι συστατικά θέλει κάθε ένας φίλος του στην πίτσα του.

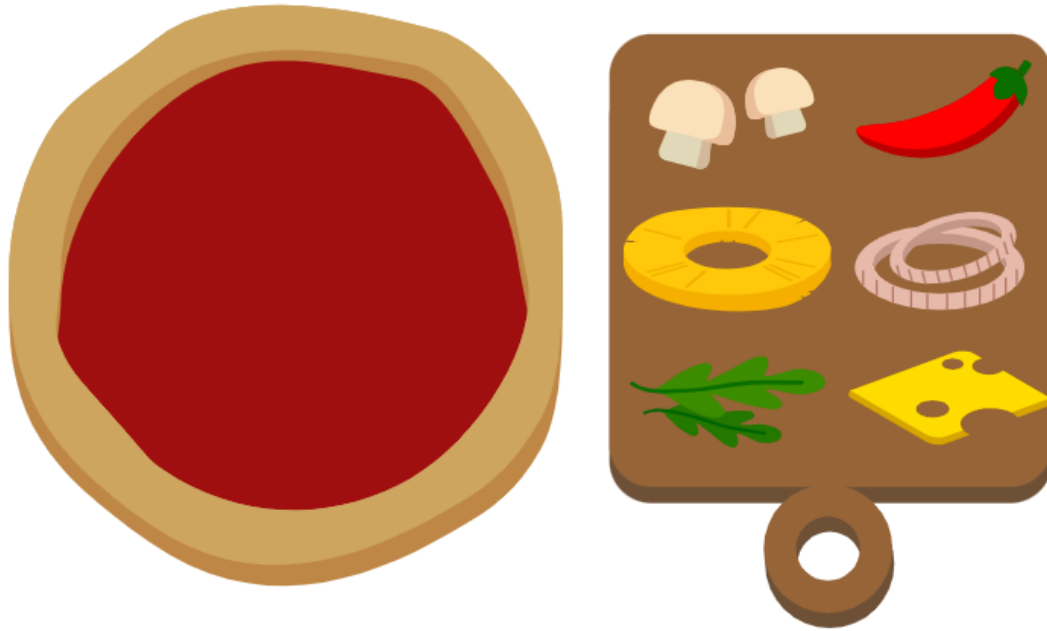
Αλίκη			
Γιώργος			
Μανώλης			
Έλενα			

Ο Γιάννης θέλει να ψήσει μια πίτσα με τρία συστατικά συνολικά. Θέλει να διαλέξει τα τρία συστατικά με τέτοιο τρόπο που να ικανοποιήσει όσο μπορεί περισσότερο τους φίλους του.

### Ερώτηση

Πάτησε στα τρία συστατικά που πρέπει να διαλέξει ο Γιάννης.

Πάτησε "Αποθήκευση" όταν έχεις τελειώσει.



## Απάντηση

Η σωστή απάντηση είναι:μανιτάρια, τυρί και κρεμμύδια.

## Εξήγηση απάντησης







Κάθε ένας από τους φίλους του Γιάννη επιθυμεί μανιτάρια (4 επιθυμίες), 3 από τους φίλους του επιθυμούν τυρί και 2 επιθυμούν κρεμμύδια. Έτσι, συνολικά 9 επιθυμίες ικανοποιούνται. Δεν υπάρχει άλλος συνδυασμός συστατικών που να μπορεί να ικανοποιήσει περισσότερες επιθυμίες.



Πως βρίσκεις τη λύση; Μπορείς απλά να δοκιμάσεις όλους τους συνδυασμούς και να μετρήσεις πόσες επιθυμίες ικανοποιούνται. Αλλά μπορείς και να υπολογίσεις τους συνδυασμούς πιο συστηματικά, όπως παρακάτω:

Βήμα 1: Για κάθε ένα συστατικό της πίτσας, καθόρισε πόσοι φίλοι το επιθυμούν.

Στον παρακάτω πίνακα φαίνονται οι επιθυμίες των συστατικών όλων των φίλων του Γιάννη. Για κάθε συστατικό, υπάρχει μια στήλη και για κάθε άτομο, υπάρχει μια σειρά. Όταν ένα άτομο επιθυμεί ένα συστατικό, τότε στον πίνακα συμπληρώνεις στην αντίστοιχη σειρά για το άτομο και στην αντίστοιχη στήλη για το συστατικό. Για παράδειγμα, η Αλίκη επιθυμεί πιπεριά έτσι στην πρώτη σειρά του πίνακα και στην πρώτη στήλη μπαίνει ο αριθμός 1.

						
Αλίκη	1	1			1	
Γιώργος		1	1			1
Μανώλης		1		1	1	
Έλενα		1			1	1
Συστατικό που είναι το αγαπημένο πολλών ατόμων	1	4	1	1	3	2

## Βήμα 2: Βρες τα τρία πιο δημοφιλή συστατικά της πίτσας

Στη συνέχεια, προσθέτεις τα νούμερα σε κάθε μια στήλη και γράφεις το τελικό αποτέλεσμα στην τελευταία σειρά. Ταμανιτάρια είναι τα πιο δημοφιλή (4), ακολουθούνται από το τυρί (3) και από τα κρεμμύδια (2). Τα υπόλοιπα συστατικά επιθυμούνται μόνο από ένα άτομο. Τα τρία πιο δημοφιλή συστατικά ικανοποιούν τις επιθυμίες των περισσότερων φίλων, 9 συνολικά επιθυμίες.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Αυτό είναι ένα παράδειγμα ενός απλού προβλήματος βελτιστοποίησης. Για να βρεις μια λύση, πρέπει να συλλέξεις δεδομένα (οι επιθυμίες των φίλων), να υπολογίσεις νέα δεδομένα (συχνότητες των επιθυμιών), να λάβεις υπόψη σου τους περιορισμούς (μόνο μία πίτσα με τρία υλικά) και να πάρεις αποφάσεις (επιλέγοντας τα υλικά). Η διαδικασία δύο βημάτων που χρησιμοποιείται στην επεξήγηση της σωστής απάντησης είναι ένας αλγόριθμος που μπορεί να λύσει αυτό το πρόβλημα βελτιστοποίησης. Είναι μια γενική λύση επειδή θα λειτουργούσε και με άλλα υλικά και άλλες επιθυμίες.

Οι αλγόριθμοι μπορούν να εκτελούνται αυτόματα από έναν υπολογιστή. Η βελτιστοποίηση παίζει ρόλο σε πολλούς τομείς της ζωής μας, όπως στις μεταφορές (βέλτιστη διαδρομή για ένα όχημα παράδοσης) ή στη βιομηχανία (βέλτιστες διαδικασίες παραγωγής).

### Υπολογιστική σκέψη

Αυτός ο γρίφος αναδεικνύει αρκετά βασικά στοιχεία της υπολογιστικής σκέψης.

Η αναγνώριση προτύπων περιλαμβάνει την αναζήτηση ομοιοτήτων ή τάσεων μέσα στο πρόβλημα. Αναλύοντας τις προτιμήσεις των συστατικών της πίτσας, μπορεί κανείς να παρατηρήσει ότι ορισμένα συστατικά είναι πιο δημοφιλή από άλλα ή ότι ορισμένοι συνδυασμοί μπορούν να ικανοποιήσουν τις προτιμήσεις πολλών ατόμων. Η αναγνώριση αυτών των προτύπων είναι κρίσιμη για τη λήψη μιας απόφασης για το ποιά συστατικά θα επιλεγούν.

Ο σχεδιασμός αλγορίθμου περιλαμβάνει τη δημιουργία μιας λύσης βήμα-βήμα για την επίλυση ενός προβλήματος. Στο γρίφο με την πίτσα, αυτό μπορεί να περιλαμβάνει μια μέθοδο για την καταμέτρηση των ψήφων για κάθε συστατικό, μια διαδικασία για την επιλογή των τριών κορυφαίων συστατικών και έναν κανόνα για το τι θα γίνει σε περίπτωση ισοπαλίας. Αυτό το βήμα αφορά τη διατύπωση μιας σαφούς, λογικής προσέγγισης για την επίτευξη του βέλτιστου συνδυασμού συστατικών για την πίτσα, δεδομένων των περιορισμών.



Σε αυτό το πρόβλημα, αν και ο αριθμός των συστατικών και των φίλων είναι σχετικά μικρός, πρέπει κανείς να αναλύσει συστηματικά τα δεδομένα (προτιμήσεις) για να πάρει μια απόφαση (ποια υλικά να επιλέξει) που επιτυγχάνει έναν στόχο (μέγιστη ικανοποίηση). Αυτή η διαδικασία θα μπορούσε να μεταφραστεί σε έναν αλγόριθμο που θα μπορούσε να εφαρμοστεί σε διαφορετικά σενάρια που να περιλαμβάνουν μεγαλύτερο αριθμό φίλων ή συστατικών, διαφορετικό αριθμό επιλεγμένων συστατικών και ακόμη και διαφορετικά κριτήρια.

Το πρόβλημα αυτό λειτουργεί ως πρακτικό παράδειγμα της υπολογιστικής σκέψης, δείχνοντας πώς αυτές οι έννοιες είναι όχι μόνο σημαντικές στον προγραμματισμό και την επιστήμη των υπολογιστών, αλλά και στην επίλυση προβλημάτων και τη λήψη αποφάσεων στην καθημερινή ζωή.

## Χώρα προέλευσης θέματος

Γερμανία

## 2024-NL-02 – Το δώρο των γενεθλίων

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Το δώρο των γενεθλίων

Η Ζωή θέλει μια μπάλα δώρο για τα γενέθλιά της.

Θέλει την μπάλα να:

- Μην έχει ρίγες
- Έχει ένα αστέρι ζωγραφισμένο σε αυτή
- Μην έχει φεγγάρι ζωγραφισμένο σε αυτή

### Ερώτηση

Ποιά μπάλα θα αγοράσουμε για τη Ζωή;



### Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Μπορούμε να χρησιμοποιήσουμε τους κανόνες που έθεσε η Ζωή για να αποφασίσουμε ποιά μπάλα θέλει.

Θέλει μια μπάλα χωρίς ρίγες, οπότε αφαιρούμε τις μπάλες που έχουν ρίγες:



Θέλει μια μπάλα που να έχει ζωγραφισμένο ένα αστέρι επάνω της, άρα αφαιρούμε όλες τις μπάλες που δεν έχουν αστέρια:

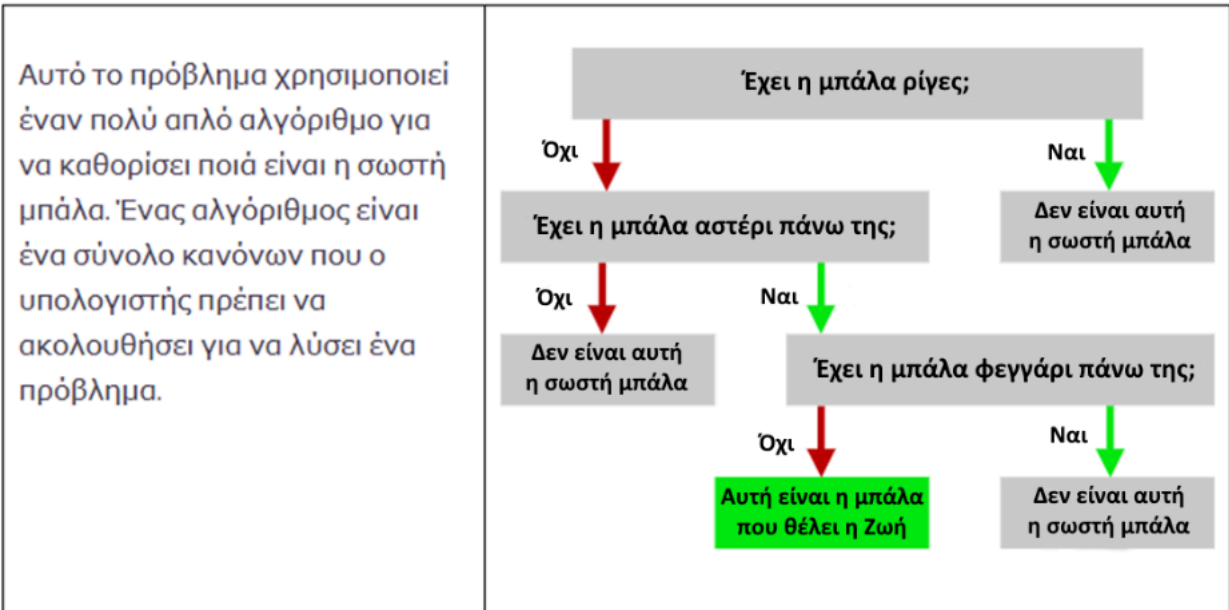


Ο τελευταίος της κανόνας είναι: να μην υπάρχει φεγγάρι ζωγραφισμένο επάνω στη μπάλα. Άρα αφαιρούμε την τελευταία μπάλα. Η μπάλα με το αστέρι είναι η μόνη μπάλα που ακολουθεί τους κανόνες της Ζωής.



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη των Υπολογιστών



### Υπολογιστική Σκέψη

Η αλγοριθμική σκέψη είναι ένας πολύ σημαντικός τρόπος για την επίλυση ενός προβλήματος που χρησιμοποιούμε στην καθημερινότητά μας. Το συγκεκριμένο πρόβλημα είναι μια εισαγωγή στον τρόπο που καθοδηγούμε τους αλγόριθμους και στον τρόπο που αντιμετωπίζουμε ένα πρόβλημα βήμα - βήμα ακολουθώντας τους κανόνες.

### Χώρα προέλευσης θέματος

Ολλανδία

## 2024-BR-04 – Οργάνωση βραχιολιών

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

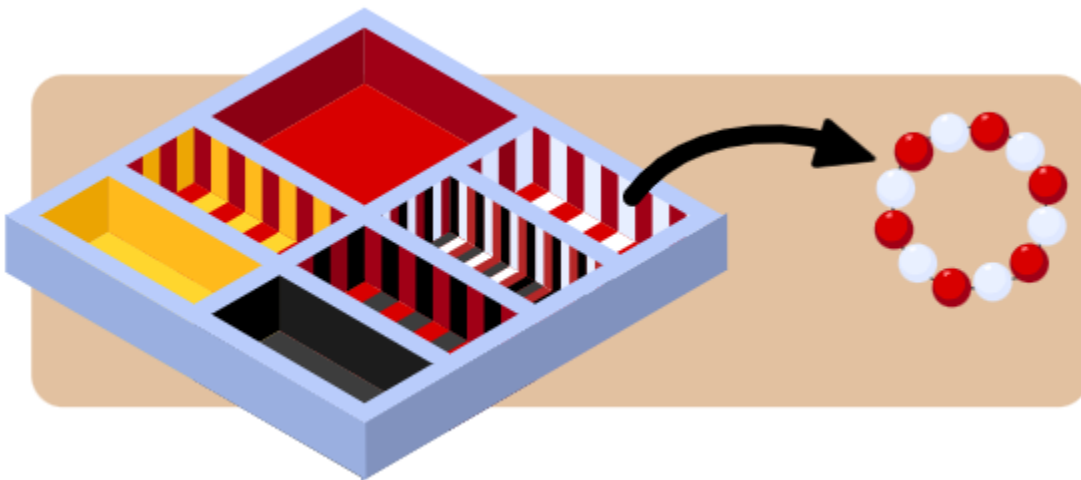
Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Οργάνωση βραχιολιών

Η Ιωάννα έχει τα βραχιόλια της σε ένα κουτί για κοσμήματα με επτά ανοικτές θέσεις.

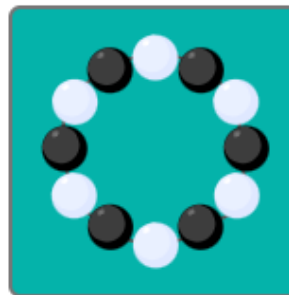
Βάζει το κάθε βραχιόλι στη θέση του κουτιού με το ίδιο χρωματικό συνδυασμό.

Αυτό είναι ένα παράδειγμα ενός βραχιολιού που ταιριάζει σε μια αντίστοιχη χρωματικά θέση του κουτιού.



Ερώτηση

Ποιό βραχιόλι δεν ταιριάζει σε καμία από τις θέσεις του κουτιού;



## Απάντηση

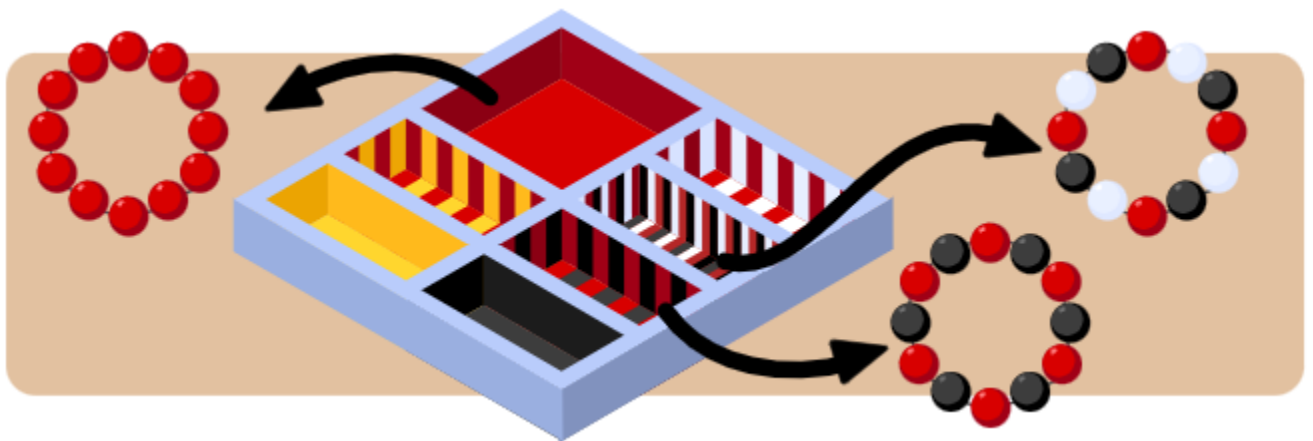


Η σωστή απάντηση είναι το βραχιόλι

## Εξήγηση απάντησης

Το βραχιόλι αυτό δεν ταιριάζει σε καμία θέση του κουτιού.

Η παρακάτω εικόνα δείχνει ποιές θέσεις ταιριάζουν τα άλλα βραχιόλια. Έχουν ένα ίδιο χρωματικό συνδυασμό.



Παρόλα αυτά, δεν υπάρχει θέση που να έχει τον χρωματικό συνδυασμό



μαύρου-άσπρου όπως το βραχιόλι

Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Συχνά, όταν ασχολούμαστε με προβλήματα πληροφορικής, έχουμε στη διάθεσή μας ένα μεγάλο όγκο δεδομένων, αλλά μόνο ένα μικρό μέρος αυτών είναι σχετικό με μια συγκεκριμένη ερώτηση. Για αυτό το πρόβλημα, χρειαζόμαστε μόνο πληροφορίες για ορισμένες ιδιότητες των βραχιολιών ώστε να τα τοποθετήσουμε στις σωστές θέσεις στο /κουτί. Οι πληροφορίες για πολλές άλλες ιδιότητες των βραχιολιών δεν βοηθούν στην επίλυση αυτού του προβλήματος. Πρέπει να κοιτάξουμε μόνο τα χρώματα, όχι το σχήμα των χαντρών ή τον αριθμό τους. Είναι σημαντικό να φιλτράρουμε τα δεδομένα που δεν χρειάζονται. Με αυτόν τον τρόπο, μπορούμε να δουλέψουμε μόνο με τα δεδομένα που θα οδηγήσουν στη λύση του προβλήματος.

Αυτή η αρχή χρησιμοποιείται επίσης στην τεχνητή νοημοσύνη, για παράδειγμα, για τη μείωση του όγκου των δεδομένων στα σχετικά σημεία κατά την αναγνώριση αντικειμένων, αυξάνοντας έτσι την υπολογιστική ισχύ.

### Υπολογιστική Σκέψη

Αυτό το πρόβλημα χρησιμοποιεί μια απλή μορφή αφαίρεσης. Πρέπει να αγνοήσουμε τις περιττές λεπτομέρειες των βραχιολιών για να βρούμε τις πληροφορίες που χρειαζόμαστε με σκοπό να λύσουμε το πρόβλημα. Σε αυτή την περίπτωση, είναι σημαντικό να συνειδητοποιήσουμε ότι το χρώμα είναι αυτό που έχει σημασία, αυτή είναι η μόνη πληροφορία που χρειαζόμαστε για τα βραχιόλια.

## Χώρα προέλευσης θέματος

Βραζιλία

## 2024-IE-01c – Περιστρεφόμενα τριφύλλια

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Περιστρεφόμενα τριφύλλια

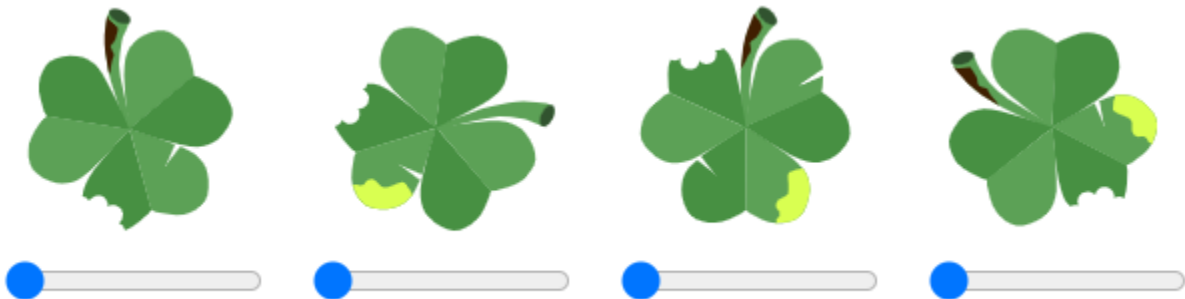
Ο Λουκάς κάνει συλλογή με τριφύλλια. Κατά λάθος, έριξε το αγαπημένο του τριφύλλι στο έδαφος. Τώρα είναι ανακατεμένο με άλλα τριφύλλια! Βοήθησέ τον να τα βρει.



### Ερώτηση

Ποιο είναι το τριφύλλι του Λουκά;

Μπορείς να περιστρέψεις το τριφύλλι μετακινώντας τον κύλινδρο. Κάνε κλικ στο σωστό τριφύλλι για να το επιλέξεις και μετά πάτησε "Αποθήκευση".





## Απάντηση



Αυτό είναι το σωστό τριφύλλι.

## Εξήγηση απάντησης

Το σωστό τριφύλλι έχει μια κίτρινη άκρη στο φύλλο του και η καφέ περιοχή του κοτσανιού βρίσκονται στην ίδια θέση με το τριφύλλι που δίνεται. Επίσης, η ραφή στο φύλλο και το κομμάτι που έχει φαγωθεί ταιριάζουν.



φύλλο.

Αυτό δεν είναι σωστό γιατί δεν έχει την κίτρινη άκρη στο πάνω



Αυτό δεν είναι σωστό γιατί δεν έχει την καφέ περιοχή στο κοτσάνι



Αυτό δεν είναι σωστό γιατί το τριφύλλι έχει σκιστεί και φαγωθεί σε διαφορετικές περιοχές.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Η πληροφορία σήμερα συχνά αναζητείται από προγράμματα υπολογιστών που αναζητούν διάφορα πρότυπα. Αναγνωρίζοντας τα πρότυπα, η πληροφορία μπορεί να φιλτραριστεί, να αναλυθεί και να κατανοηθεί καλύτερα. Αφαιρώντας ό,τι δεν ταιριάζει με το πρότυπο, μπορούμε συχνά να βρούμε καλύτερα αποτελέσματα.

Τα πρότυπα μπορούν να υπάρχουν σε οποιοδήποτε είδος δεδομένων, χωρικά και χρονικά, είτε ως τέλεια αντίγραφα είτε με σφάλματα, ή ακόμα και μετασχηματισμένα (π.χ. αντεστραμμένα). Τα πρότυπα μπορούν επίσης να εμφανιστούν σε διαφορές μεταξύ συνόλων δεδομένων. Τα πρότυπα σε αυτό το πρόβλημα είναι ελαφρώς κρυμμένα λόγω περιστροφής και αναστροφής. Αυτό συμβαίνει με σκοπό να δείξουμε ότι τα δεδομένα μπορεί να χρειαστεί να υποστούν επεξεργασία για να αποκαλυφθεί το πρότυπο που αναζητούμε.

Αναγνώριση προτύπων. Αυτό το πρόβλημα δείχνει την έννοια της εύρεσης παρόμοιων στοιχείων. Κάθε στοιχείο του τριφυλλιού (π.χ. το μικρό σχίσσιμο απέναντι από το σημάδι του δαγκώματος) μπορεί να χρησιμοποιηθεί ως βάση για την αναζήτηση παρόμοιων στοιχείων από τη λίστα των απαντήσεων. Η μη εύρεση παρόμοιου στοιχείου είναι αρκετή για να αποκλείσουμε μία από τις απαντήσεις.

Αφαίρεση. Αυτό το πρόβλημα αναδεικνύει την έννοια της δημιουργίας ενός αφαιρετικού μοντέλου. Για να λύσουμε αυτό το πρόβλημα μπορούμε να σημειώσουμε τα χαρακτηριστικά του τριφυλλιού που δεν αλλάζουν όταν το τριφύλλι περιστρέφεται (π.χ. ένα χρήσιμο χαρακτηριστικό θα ήταν η κίτρινη άκρη στο ίδιο φύλλο που έχει δαγκωθεί). Ένα μικρό σύνολο τέτοιων χαρακτηριστικών (ένα αφαιρετικό μοντέλο· σε αυτή την περίπτωση μια συμβολική αναπαράσταση της εικόνας) θα μπορούσε να χαρακτηρίσει μοναδικά το τριφύλλι, ενώ όλα τα υπόλοιπα στοιχεία μπορούν να αγνοηθούν. Υπάρχουν πολλές εξίσου έγκυρες πιθανότητες για ένα τέτοιο μικρό σύνολο χαρακτηριστικών. Το σύνολο αυτών των χαρακτηριστικών μπορεί να συγκριθεί με κάθε πολλαπλή επιλογή για να βρεθεί το τριφύλλι που ταιριάζει.

## Χώρα προέλευσης θέματος

Ιρλανδία

## 2024-PL-03 – Σχεδιάζοντας ένα ιστιοφόρο

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Σχεδιάζοντας ένα ιστιοφόρο

Η Σοφία θέλει να σχεδιάσει το παρακάτω ιστιοφόρο.

Σκέφτεται τους δύο παρακάτω κανόνες:

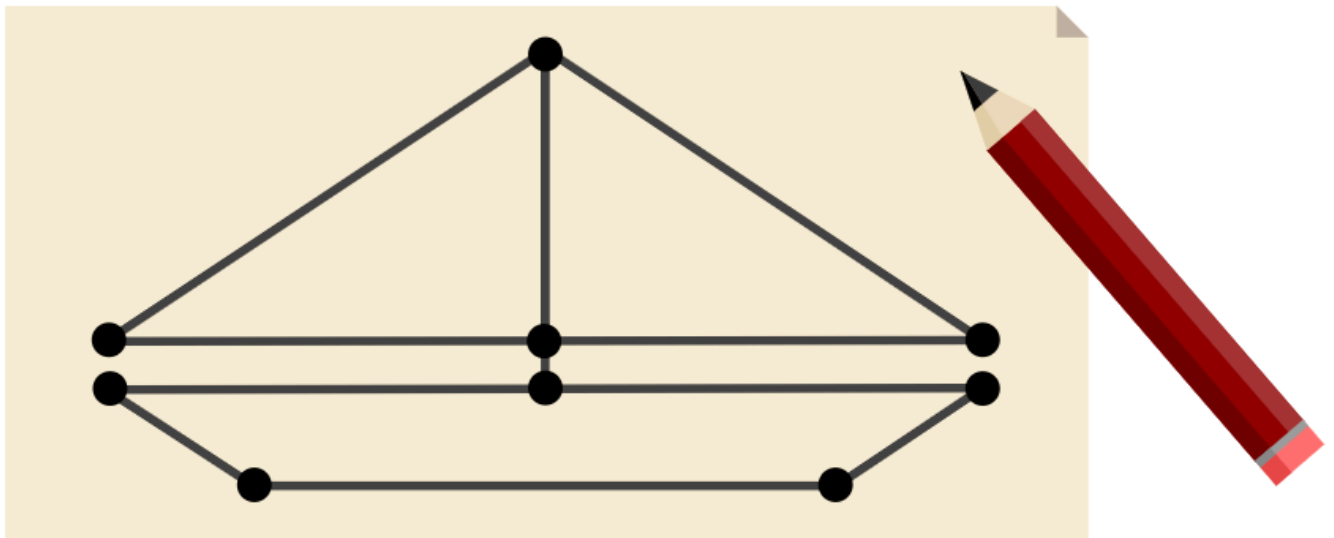
Το ιστιοφόρο πρέπει να σχεδιαστεί με μια γραμμή συνεχόμενη. Η Σοφία δεν πρέπει να σηκώσει το μολύβι από το χαρτί.

Μια γραμμή μπορεί να σχεδιαστεί μόνο μια φορά. Δεν μπορεί να ξανασχεδιαστεί πολλές φορές.

### Ερώτηση

Από πού πρέπει να αρχίσει η Σοφία τη σχεδίαση; Πάτησε στο σημείο της αρχής.

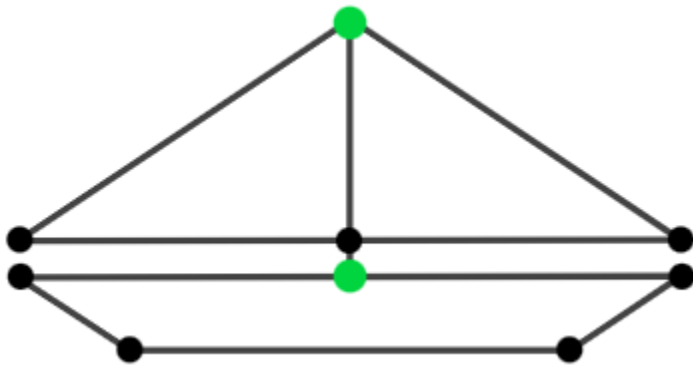
*Πάτησε "Αποθήκευση" όταν τελειώσεις.*



## Απάντηση

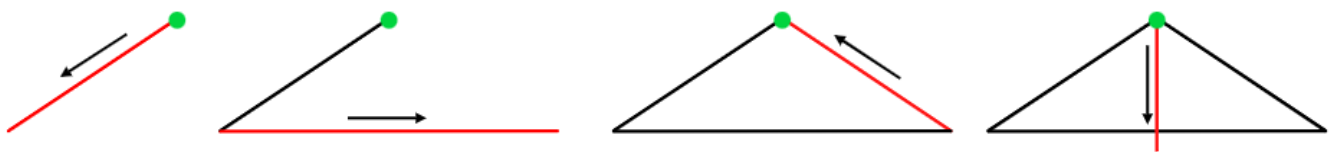
Η Σοφία μπορεί να βρει ποιιά σημεία είναι κατάλληλα ως σημεία εκκίνησης, σχεδιάζοντας τις γραμμές από κάθε έναν από τους εννέα πράσινους κύκλους.

Υπάρχουν δύο επιτρεπτά σημεία εκκίνησης.

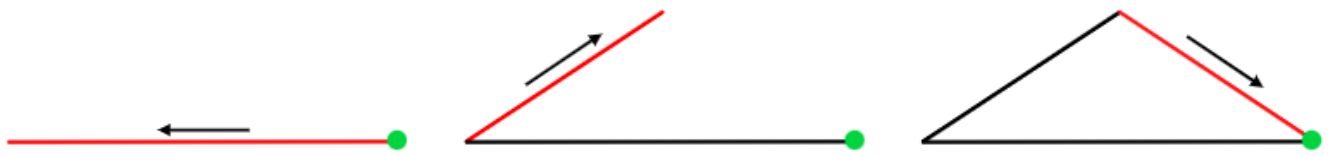


## Εξήγηση απάντησης

Και τα δύο αυτά σημεία έχουν έναν περιττό αριθμό γραμμών που εξέρχονται από αυτά. Όταν ξεκινάμε από ένα σημείο που έχει περιττό αριθμό εξερχόμενων γραμμών και σχεδιάζουμε την πρώτη γραμμή, πρέπει να επιστρέψουμε στο ίδιο σημείο για να σχεδιάσουμε τη δεύτερη γραμμή. Στη συνέχεια, χρησιμοποιούμε την τρίτη γραμμή για να αλλάξουμε κατεύθυνση ώστε να εξέλθουμε ξανά από το σημείο και να ολοκληρώσουμε το σχέδιο χωρίς να περάσουμε από τις προηγούμενες δύο γραμμές.



Όλες οι λανθασμένες απαντήσεις είναι σημεία με ζυγό αριθμό γραμμών που εξέρχονται από αυτά. Διότι κάθε φορά που βγαίνουμε από ένα σημείο και επιστρέφουμε σε αυτό, έχουμε σχεδιάσει δύο γραμμές από αυτό (ζυγός αριθμός), και δεν είναι δυνατό να εξέλθουμε ξανά από αυτό χωρίς μια επιπλέον γραμμή (περιττός αριθμός γραμμών).



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Το σχέδιο του ιστιοφόρου αποτελείται από σημεία και γραμμές. Στην επιστήμη των υπολογιστών, αυτό ονομάζεται γράφημα. Ένα γράφημα αποτελείται από κόμβους και ακμές. Οι κόμβοι είναι τα σημεία και οι ακμές είναι οι γραμμές που συνδέουν δύο κόμβους.

Στο πρόβλημα αυτό, υπάρχει η ανάγκη να βρεθεί μια διαδρομή που περνά από όλους τους κόμβους και στην οποία όλες οι ακμές εμφανίζονται ακριβώς μία φορά. Οι κόμβοι μπορούν να χρησιμοποιηθούν όσες φορές θέλετε. Για να μπορεί να λυθεί το πρόβλημα, δεν πρέπει να υπάρχουν κόμβοι που να μην μπορούν να χρησιμοποιηθούν καθόλου. Ένα γράφημα που πληροί αυτή την προϋπόθεση λέγεται 'συνδεδεμένο'. Το «Σπίτι του Άγιου Βασίλη» είναι ένας γνωστός γρίφος στον οποίο παιδιά και ενήλικες πρέπει να βρουν μια διαδρομή σε ένα συνδεδεμένο γράφημα.

## Χώρα προέλευσης θέματος

Πολωνία









## 2024-KR-02 – Πάρτυ στην αυλή

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Πάρτυ στην αυλή

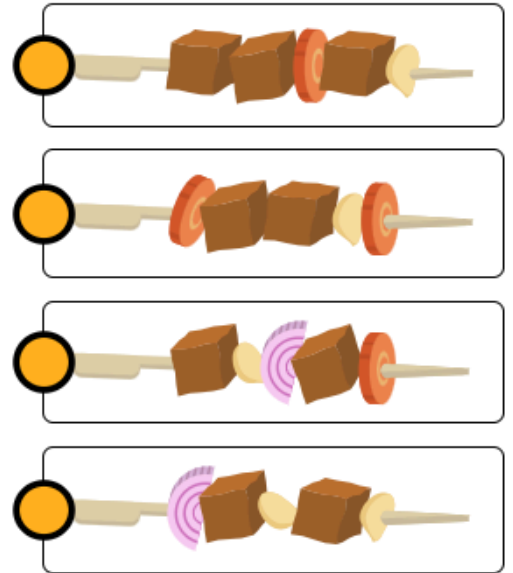
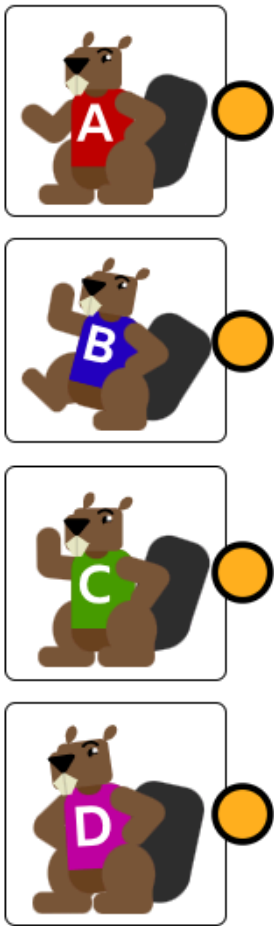
Στο πάρτυ στην αυλή, τέσσερις κάστορες περιγράφουν πως θέλουν να είναι το σουβλάκι τους.

			
Χρειάζομαι 	Δεν τρώω 	Όσο περισσότερο  γίνεται	Όσο περισσότερο  γίνεται

### Ερώτηση

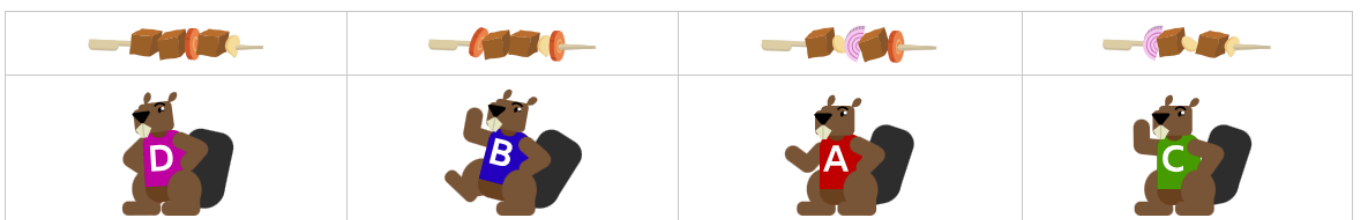
Υπάρχουν τέσσερα σουβλάκια. Μοίρασε τα ώστε όλοι οι κάστορες να είναι όσο το δυνατόν ικανοποιημένοι.

Σχεδίασε μια γραμμή μεταξύ των καστόρων και του κάθε ενός σουβλακιού για να τα ενώσεις. Πάτησε "Αποθήκευση" όταν τελειώσεις.



## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Ας αριθμήσουμε τα σουβλάκια:



Μόνο το 4ο σουβλάκι έχει παραπάνω από ένα κομμάτι σκόρδο. Ο κάστορας με το γράμμα C παίρνει το 4ο σουβλάκι.

Μόνο το 1ο σουβλάκι έχει περισσότερα από ένα κομμάτια κρέας. Ο κάστορας με το γράμμα D παίρνει το 1ο σουβλάκι.

Από τα δύο σουβλάκι που απομένουν, μόνο το 3ο έχει κρεμμύδι, αυτό λοιπόν πρέπει να το πάρει ο κάστορας με το γράμμα A και όχι με το γράμμα B.

Ο κάστορας με το γράμμα B παίρνει το 2ο σουβλάκι.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Το Πρόβλημα Ικανοποίησης Περιορισμών (Constraint Satisfaction Problem - CSP) είναι μία από τις βασικές έννοιες στην επιστήμη των υπολογιστών. Ένα CSP είναι το πρόβλημα εύρεσης μιας τιμής που ικανοποιεί ορισμένους περιορισμούς σε ένα σύνολο μεταβλητών. Αυτά τα προβλήματα συναντώνται σε διάφορους τομείς, όπως ο προγραμματισμός μιας εργασίας, ο σχεδιασμός, η κατανομή πόρων, ο σχεδιασμός μιας διάταξης και η υπολογιστική όραση. Στην πληροφορική, το CSP χρησιμοποιείται ως μια σημαντική προσέγγιση για την επίλυση προβλημάτων. Η πληροφορική περιλαμβάνει την ανάλυση δεδομένων, την ανάπτυξη αλγορίθμων και τον σχεδιασμό λογισμικού και συστημάτων, όπου το CSP μπορεί να αξιοποιηθεί για την ανάπτυξη αποδοτικών λύσεων. Για παράδειγμα, το CSP μπορεί να χρησιμοποιηθεί σε συστήματα βάσεων δεδομένων όταν χρειάζεται να οριστούν περιορισμοί για τη διατήρηση της συνέπειας των δεδομένων και να βρεθούν λύσεις που να τους ικανοποιούν.

### Υπολογιστική Σκέψη



Αυτό το πρόβλημα είναι ένα πρόβλημα ικανοποίησης περιορισμών, που σημαίνει ότι για να το λύσετε, πρέπει να οργανώσετε τις συνθήκες που θέλει κάθε κάστορας, εφαρμόζοντας λογική σκέψη. Στη συνέχεια, πρέπει να δούμε τις επιλογές που έχουν τους περιορισμούς. Υποθέτοντας ότι μπορείτε να ικανοποιήσετε όλους τους κάστορες, ξεκινάτε ικανοποιώντας πρώτα εκείνον που έχει τους πιο αυστηρούς περιορισμούς. Έπειτα, ικανοποιείτε την επόμενη πιο αυστηρή συνθήκη, και συνεχίζετε έτσι μέχρι να ικανοποιηθούν όλες οι συνθήκες. Αυτή η διαδικασία επιτρέπει στους μαθητές να βιώσουν τη διαδικασία διάσπασης ενός σύνθετου προβλήματος σε μικρότερα υπο-προβλήματα, να ορίσουν περιορισμούς για κάθε υπο-πρόβλημα και στη συνέχεια να βρουν μια λύση που να ικανοποιεί αυτούς τους περιορισμούς.

## Χώρα προέλευσης θέματος

Νότια Κορέα

## 2024-MY-03 – Διαδικτυακή τάξη

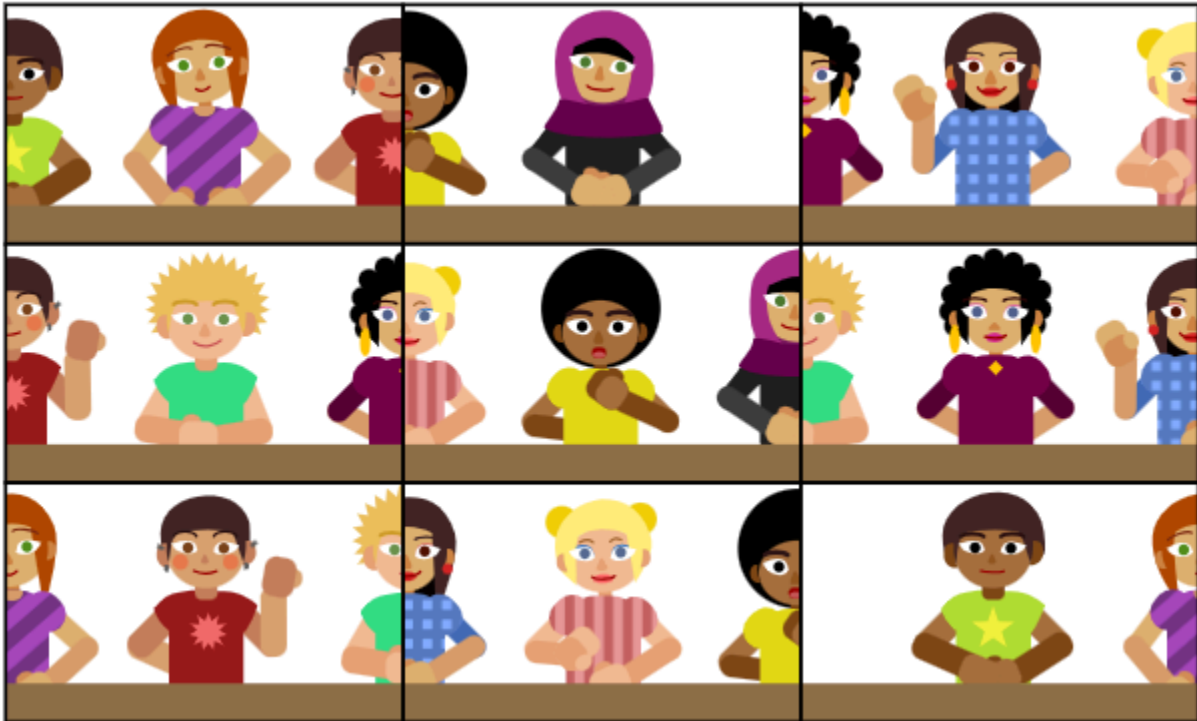
Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Διαδικτυακή τάξη

Η Αναστασία πηγαίνει σε ένα νέο σχολείο αλλά θέλει να δει τους φίλους της από το παλιό σχολείο ξανά. Οι φίλοι της κάθονται ο ένας δίπλα στον άλλο αλλά σε διαφορετικούς ηλεκτρονικούς υπολογιστές.

Έτσι βλέπει η Αναστασία τους φίλους της στην οθόνη του υπολογιστή της:



### Ερώτηση

Σύρε τους φίλους της Αναστασίας τον ένα δίπλα στον άλλο σαν να κάθονταν σε μια σειρά.

Σύρε τους φίλους στη σωστή σειρά και μετά πάτησε "Αποθήκευση".




## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Με βάση την εικόνα που βλέπουμε στην οθόνη του υπολογιστή, γνωρίζουμε ποιός κάθεται δίπλα σε ποιόν.

<p>Από την εικόνα στην κάτω δεξιά πλευρά της οθόνης ξέρουμε ότι ο James έχει μόνο έναν δίπλα του. Έτσι καταλαβαίνουμε ότι κάθεται στη μια άκρη της σειράς.</p>	
<p>Στην εικόνα πάνω αριστερά βλέπουμε ότι η Emma κάθεται δίπλα του και δίπλα στην Emma κάθεται η Diana.</p>	

Εάν εφαρμόσεις την ίδια μέθοδο στις άλλες εικόνες, μπορείς να δεις ότι οι 9 φίλοι κάθονται με την παραπάνω σειρά.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Σε αυτό το πρόβλημα, έχουμε εικόνες όπου βλέπουμε όχι μόνο ένα παιδί αλλά και μια εικόνα από τα παιδιά που κάθονται δίπλα του. Αυτά λειτουργούν ως "αναφορές" στους γείτονες.

Οι επιστήμονες υπολογιστών χρησιμοποιούν παρόμοιες δομές δεδομένων.

Αυτή η δομή δεδομένων, όπου είναι τοποθετημένοι οι φίλοι, ονομάζεται "διπλά συνδεδεμένη λίστα". Σε μια διπλά συνδεδεμένη λίστα, κάθε κόμβος της λίστας περιέχει το στοιχείο, μια αναφορά στον προηγούμενο κόμβο και μια αναφορά στον επόμενο κόμβο. Σε αυτό το πρόβλημα, κάθε οθόνη αντιστοιχεί σε έναν κόμβο, το παιδί είναι το στοιχείο της λίστας και υπάρχουν αναφορές (δείκτες) προς τα παιδιά στα δεξιά και τα αριστερά του παιδιού. Αυτό το πρόβλημα περιλαμβάνει την πρακτική πλοήγησης σε μια λίστα, ξεκινώντας από οποιοδήποτε άκρο. Αυτή η διπλή σύνδεση βοηθά τους επιστήμονες υπολογιστών να φτάνουν όχι μόνο στο επόμενο αλλά και στο προηγούμενο στοιχείο. Έτσι, μπορούν να βρουν και να επεξεργαστούν ένα στοιχείο πιο γρήγορα. Φυσικά, χρειάζεται περισσότερη μνήμη για να αποθηκευτούν δύο αναφορές για όλα τα στοιχεία.

Η υπολογιστική σκέψη σε αυτό το πρόβλημα αναφέρεται στη χρήση λογικής σκέψης για την επίλυσή της. Αυτό περιλαμβάνει την κατανόηση του προβλήματος, τη δημιουργία μιας αλγοριθμικής προσέγγισης για την αξιολόγηση των συνθηκών και την εφαρμογή της λογικής στο δεδομένο σενάριο.

Η λογική σκέψη είναι απαραίτητη για την εξεύρεση της λύσης. Απαιτεί την ανάλυση του προβλήματος σε βήματα για να βρεθεί μια λύση. Αν έχετε κοιτάξει ποτέ οδηγούς για τη συναρμολόγηση γρίφων, μπορεί να βρείτε συγκεκριμένα μοτίβα σύνδεσης κομματιών, να εντοπίσετε τις άκρες, να ταξινομήσετε κατά σχήμα και να δουλέψετε τμήμα προς τμήμα.

## Χώρα προέλευσης θέματος

Μαλαισία

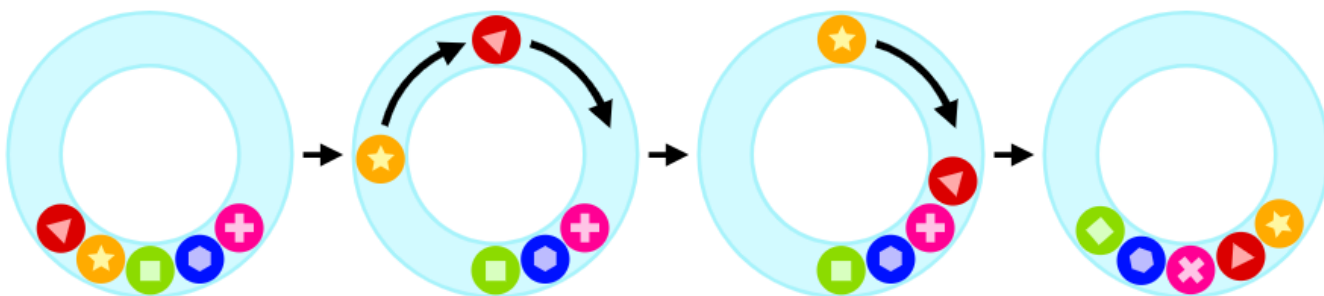
## 2024-SK-01b – Η κουδουνίστρα του Άλκη

Ηλικιακή ομάδα: Α' – Β' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Η κουδουνίστρα του Άλκη

Ο Άλκης έχει μια διαφανή κουδουνίστρα με πολύχρωμες μπάλες. Όταν την κουνάει, μερικές μπάλες περνούν μέσα από την κουδουνίστρα και μετακινούνται στην άλλη πλευρά, όπως μπορείτε να δείτε στην εικόνα.



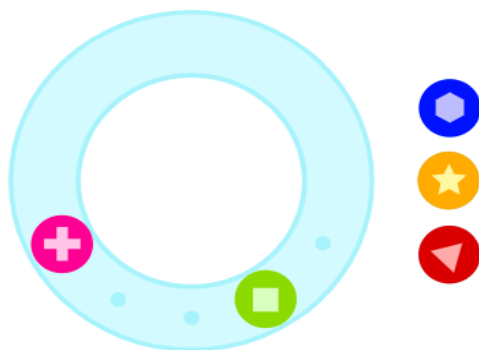
Ο Άλκης κουνάει την κουδουνίστρα άλλη μια φορά.

### Ερώτηση

Πώς μοιάζει η κουδουνίστρα του Άλκη αφού την κουνήσει;

Σύρετε τις μπάλες στη σωστή θέση μέσα στην κουδουνίστρα.

Κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.



## Απάντηση



Όταν οι μπάλες μπουν μέσα στην κουδουνίστρα, δεν μπορείς να αλλάξεις τη σειρά τους. Άρα χρειάζεται μόνο να κοιτάξεις τη σειρά όλων των μπαλών. Μετά την κόκκινη μπάλα



έχει σειρά η κίτρινη



. Μετά την πράσινη μπάλα



είναι η σειρά της μπλε



. Στη συνέχεια παίρνει σειρά η μοβ



και μετά η κόκκινη ξανά



## Εξήγηση απάντησης

Μετά την πράσινη μπάλα, μια μπάλα λείπει, μεταξύ της πράσινης και της μοβ. Σχετικά με την προηγούμενη παράγραφο αυτή πρέπει να είναι η μπλε μπάλα.

Μπορούμε να δούμε ότι μεταξύ της μοβ και της πράσινης μπάλας υπάρχουν δύο μπάλες που λείπουν. Αυτό μας αφήνει με τις επιλογές μιας κίτρινης και μιας κόκκινης μπάλας. Στη σειρά στην αρχή, παρατηρούμε ότι πριν την πράσινη μπάλα πρέπει να υπάρχει μια κίτρινη. Και μεταξύ της μοβ και της κίτρινης μπάλας πρέπει να υπάρχει μια κόκκινη μπάλα.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Οι επιστήμονες υπολογιστών αναπαριστούν και αποθηκεύουν τα δεδομένα με διάφορους τρόπους. Όλες οι αναπαραστάσεις δεδομένων έχουν πλεονεκτήματα που εξαρτώνται από τις ιδιότητες των αντίστοιχων δεδομένων.

Σε αυτό το πρόβλημα χρησιμοποιείται η λεγόμενη κυκλική συνδεδεμένη λίστα, η οποία ονομάζεται "κουδουνίστρα" και τα αποθηκευμένα στοιχεία είναι οι μπάλες. Σε αυτή την κυκλική-συνδεδεμένη λίστα, κάθε στοιχείο έχει ένα λεγόμενο δείκτη προς το επόμενο στοιχείο, δηλαδή δείχνει πιο είναι το επόμενο στοιχείο. Επιπλέον, ο δείκτης του τελευταίου στοιχείου δείχνει στο πρώτο στοιχείο και έτσι κλείνει τον κύκλο.

Αυτό σημαίνει ότι, σε αυτό το πρόβλημα, γνωρίζεις για κάθε μπάλα ποια μπάλα είναι η επόμενη. Μετά την τελευταία μπάλα έρχεται η πρώτη ξανά. Όταν ελέγχεις τη σειρά των μπαλών, μπορείς να ξεκινήσεις από οποιαδήποτε μπάλα και πρέπει να διασχίσεις όλες (ολόκληρη τη λίστα) με τη σωστή σειρά.

Υπάρχουν δύο πλεονεκτήματα όταν χρησιμοποιείς μια κυκλική συνδεδεμένη λίστα για την αποθήκευση στοιχείων. Το πρώτο πλεονέκτημα είναι ότι σε μια κυκλική συνδεδεμένη λίστα δεν υπάρχει τελικό σημείο. Η λίστα επαναλαμβάνεται πάλι από την αρχή, και έτσι τα στοιχεία ποτέ δεν τελειώνουν. Το δεύτερο πλεονέκτημα είναι η αποδοτικότητα της πλοήγησης μέσα στη λίστα. Δηλαδή είναι εύκολο να διατρέξεις τη λίστα χωρίς να χρειάζεται να ακολουθήσεις ειδικούς κανόνες για το τελικό σημείο. Απλώς συνεχίζεις προς μία κατεύθυνση.

Υπολογιστική σκέψη

Σε αυτό το πρόβλημα, προσπάθησες να αναζητήσεις ένα μοτίβο. Αυτό ονομάζεται αναγνώριση μοτίβου. Η αναγνώριση μοτίβων είναι χρήσιμη για να καθορίσεις αν κάποια προβλήματα είναι παρόμοια ή ακόμη και τα ίδια. Τότε, δεν χρειάζεται να λύσεις το καθένα ξεχωριστά, αντίθετα μπορείς να χρησιμοποιήσεις τη λύση σε ένα από αυτά για να σε βοηθήσει να λύσεις το άλλο.

## Χώρα προέλευσης θέματος

Σλοβακία






## 2024-ID-04 – Ο κάστορας χάθηκε

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Ο κάστορας χάθηκε

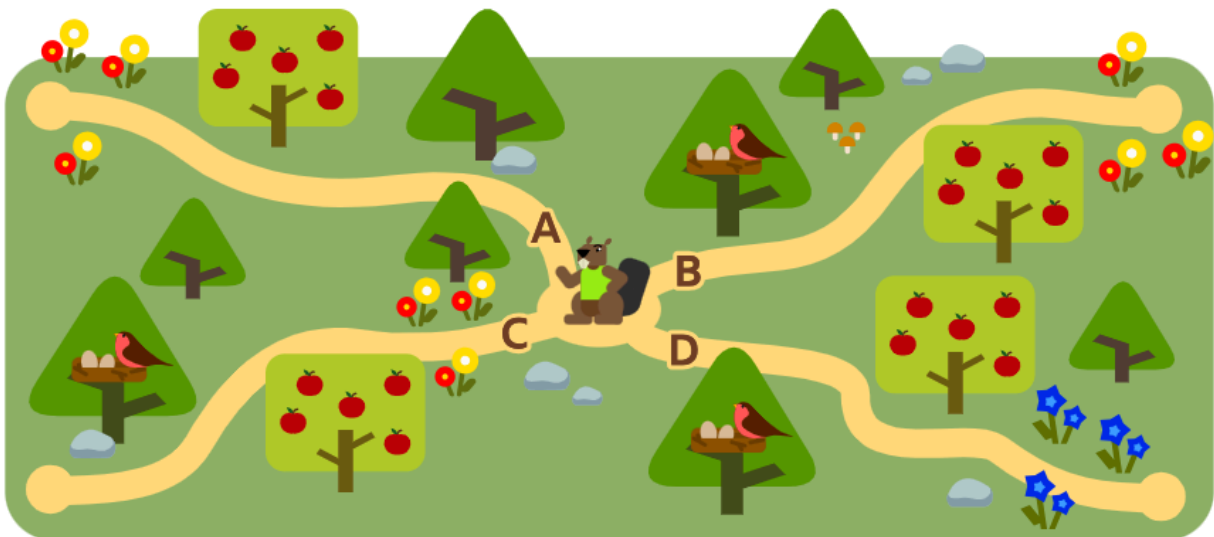
Ο Κάστορας Ελένη είναι στο δάσος. Στο δρόμο συνάντησε τα εξής:

Πρώτα, είδε κόκκινα και κίτρινα λουλούδια.	
Στην συνέχεια, είδε μια μηλιά.	
Τέλος, είδε μια φωλιά πουλιού σε ένα δέντρο.	

### Ερώτηση

Κάνε κλικ στο γράμμα στο μονοπάτι που χρησιμοποιεί η Ελένη για να μπει στο δάσος.

Κάνε κλικ στην επιλογή "Αποθήκευση" όταν τελειώσεις.



## Απάντηση

Η σωστή απάντηση είναι το μονοπάτι B.







## Εξήγηση απάντησης

Στο δρόμο προς το δάσος, η Ελένη βλέπει, πρώτα, κόκκινα και κίτρινα λουλούδια, έπειτα, βλέπει μια μηλιά και στο τέλος, μια φωλιά πουλιών σε ένα δέντρο. Στο δρόμο για το σπίτι, βλέπει τα ίδια αντικείμενα, αλλά με την αντίθετη σειρά. Έτσι, στο δρόμο για το σπίτι, βλέπει πρώτα μια φωλιά πουλιών, μετά μια μηλιά και τέλος τα κόκκινα και κίτρινα λουλούδια. Ως εκ τούτου, το B είναι η σωστή απάντηση.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Αυτό το πρόβλημα προσομοιώνει μια δομή δεδομένων στοίβας (stack) που χρησιμοποιεί τη διαδικασία LIFO (Last In First Out), δηλαδή "Τελευταίο που μπαίνει, πρώτο που βγαίνει". Τα αντικείμενα που βλέπει η Ελένη μπορούν να θεωρηθούν ως αντικείμενα που τοποθετούνται σε μια στοίβα. Το πρώτο αντικείμενο που είδε (κόκκινα και κίτρινα λουλούδια) τοποθετείται σε μια στοίβα, ακολουθούμενο από το δεύτερο αντικείμενο (μηλιά) και το τρίτο αντικείμενο (φωλιά πουλιού πάνω στο δέντρο), έτσι ώστε η στοίβα να περιγράφεται ως εξής:

Το πρώτο που είδε η Ελένη	Το δεύτερο που είδε η Ελένη	Το τρίτο που είδε η Ελένη
		
		
		

Όταν η Ελένη περπατά προς την αντίθετη κατεύθυνση, κάθε αντικείμενο στην στοίβα πρέπει να αφαιρεθεί. Το πρώτο πράγμα που θα δει στον δρόμο της επιστροφής η Ελένη είναι μια φωλιά πουλιού σε ένα δέντρο. Στη συνέχεια, η φωλιά αφαιρείται από την στοίβα, ώστε στην κορυφή της στοίβας να βρίσκεται η μηλιά. Έπειτα, αφαιρείται και η μηλιά από την στοίβα. Το τελευταίο αντικείμενο που αφαιρείται από την στοίβα είναι τα κόκκινα και κίτρινα λουλούδια.

#### Υπολογιστική σκέψη

Για να λύσουμε αυτό το πρόβλημα πρέπει να επικεντρωθούμε σε ορισμένα πράγματα και να αγνοήσουμε άλλα. Χρειάζεται να δώσουμε προσοχή μόνο στα κόκκινα και κίτρινα λουλούδια, τη μηλιά και τη φωλιά του πουλιού. Όλα τα υπόλοιπα δεν μας βοηθούν να λύσουμε το πρόβλημα, οπότε μπορούμε να τα παραβλέψουμε. Αυτό ονομάζεται αφαίρεση.

#### Χώρα προέλευσης θέματος

Ινδονησία

## 2024-ΙΕ-01c – Περιστρεφόμενα τριφύλλια

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Περιστρεφόμενα τριφύλλια

Ο Λουκάς κάνει συλλογή με τριφύλλια. Κατά λάθος, έριξε το αγαπημένο του τριφύλλι στο έδαφος. Τώρα είναι ανακατεμένο με άλλα τριφύλλια! Βοήθησέ τον να τα βρει.



### Ερώτηση

Ποιο είναι το τριφύλλι του Λουκά;

Μπορείς να περιστρέψεις το τριφύλλι μετακινώντας τον κύλινδρο. Κάνε κλικ στο σωστό τριφύλλι για να το επιλέξεις και μετά πάτησε "Αποθήκευση".



## Απάντηση



Αυτό είναι το σωστό τριφύλλι.

## Εξήγηση απάντησης

Το σωστό τριφύλλι έχει μια κίτρινη άκρη στο φύλλο του και η καφέ περιοχή του κοτσανιού βρίσκονται στην ίδια θέση με το τριφύλλι που δίνεται. Επίσης, η ραφή στο φύλλο και το κομμάτι που έχει φαγωθεί ταιριάζουν.



φύλλο.

Αυτό δεν είναι σωστό γιατί δεν έχει την κίτρινη άκρη στο πάνω



Αυτό δεν είναι σωστό γιατί δεν έχει την καφέ περιοχή στο κοτσάνι



Αυτό δεν είναι σωστό γιατί το τριφύλλι έχει σκιστεί και φαγωθεί σε διαφορετικές περιοχές.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Η πληροφορία σήμερα συχνά αναζητείται από προγράμματα υπολογιστών που αναζητούν διάφορα πρότυπα. Αναγνωρίζοντας τα πρότυπα, η πληροφορία μπορεί να φιλτραριστεί, να αναλυθεί και να κατανοηθεί καλύτερα. Αφαιρώντας ό,τι δεν ταιριάζει με το πρότυπο, μπορούμε συχνά να βρούμε καλύτερα αποτελέσματα.

Τα πρότυπα μπορούν να υπάρχουν σε οποιοδήποτε είδος δεδομένων, χωρικά και χρονικά, είτε ως τέλεια αντίγραφα είτε με σφάλματα, ή ακόμα και μετασχηματισμένα (π.χ. αντεστραμμένα). Τα πρότυπα μπορούν επίσης να εμφανιστούν σε διαφορές μεταξύ συνόλων δεδομένων. Τα πρότυπα σε αυτό το πρόβλημα είναι ελαφρώς κρυμμένα λόγω περιστροφής και αναστροφής. Αυτό συμβαίνει με σκοπό να δείξουμε ότι τα δεδομένα μπορεί να χρειαστεί να υποστούν επεξεργασία για να αποκαλυφθεί το πρότυπο που αναζητούμε.

Αναγνώριση προτύπων. Αυτό το πρόβλημα δείχνει την έννοια της εύρεσης παρόμοιων στοιχείων. Κάθε στοιχείο του τριφυλλιού (π.χ. το μικρό σχίσσιμο απέναντι από το σημάδι του δαγκώματος) μπορεί να χρησιμοποιηθεί ως βάση για την αναζήτηση παρόμοιων στοιχείων από τη λίστα των απαντήσεων. Η μη εύρεση παρόμοιου στοιχείου είναι αρκετή για να αποκλείσουμε μία από τις απαντήσεις.

Αφαίρεση. Αυτό το πρόβλημα αναδεικνύει την έννοια της δημιουργίας ενός αφαιρετικού μοντέλου. Για να λύσουμε αυτό το πρόβλημα μπορούμε να σημειώσουμε τα χαρακτηριστικά του τριφυλλιού που δεν αλλάζουν όταν το τριφύλλι περιστρέφεται (π.χ. ένα χρήσιμο χαρακτηριστικό θα ήταν η κίτρινη άκρη στο ίδιο φύλλο που έχει δαγκωθεί). Ένα μικρό σύνολο τέτοιων χαρακτηριστικών (ένα αφαιρετικό μοντέλο· σε αυτή την περίπτωση μια συμβολική αναπαράσταση της εικόνας) θα μπορούσε να χαρακτηρίσει μοναδικά το τριφύλλι, ενώ όλα τα υπόλοιπα στοιχεία μπορούν να αγνοηθούν. Υπάρχουν πολλές εξίσου έγκυρες πιθανότητες για ένα τέτοιο μικρό σύνολο χαρακτηριστικών. Το σύνολο αυτών των χαρακτηριστικών μπορεί να συγκριθεί με κάθε πολλαπλή επιλογή για να βρεθεί το τριφύλλι που ταιριάζει.

## Χώρα προέλευσης θέματος

Ιρλανδία

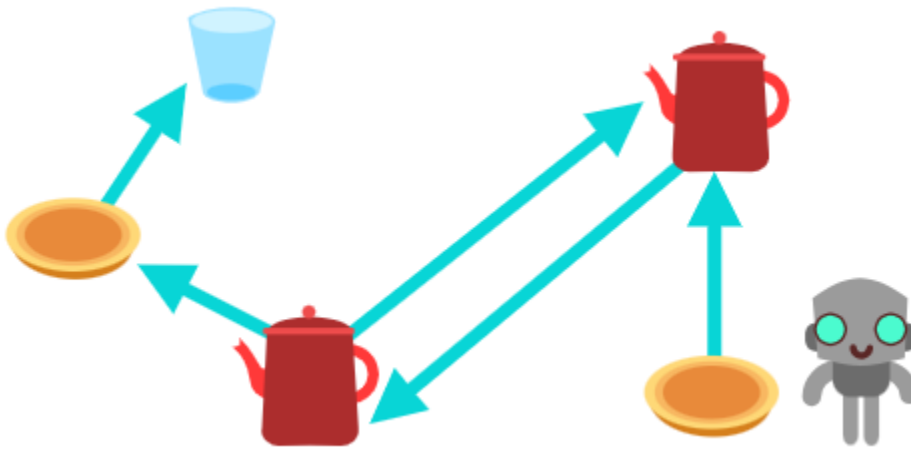
## 2024-LT-01 – Σερβίτσιο στο ράφι

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Σερβίτσιο στο ράφι

Ένα ρομπότ τοποθετεί το σερβίτσιο στα ράφια της κουζίνας. Εάν χρησιμοποιήσει τους κανόνες που φαίνονται στο παρακάτω σχήμα:



### Ερώτηση

Σε ποιο ράφι είναι το σερβίτσιο που δεν είναι στη σωστή σειρά;

Πάτησε στο γράμμα δίπλα στο σωστό ράφι. Πάτησε "Αποθήκευση" όταν τελειώσεις.



## Απάντηση

Η απάντηση είναι το γράμμα C.

## Εξήγηση απάντησης

Παρατηρήσατε ότι ο αριθμός των τσαγιερών σε μια σειρά στο ράφι πρέπει να είναι ζυγός; Αυτό συμβαίνει επειδή το διάγραμμα δείχνει επανάληψη με τις τσαγιέρες: όταν η πρώτη τσαγιέρα τοποθετείται, πρέπει να τοποθετηθεί και η δεύτερη τσαγιέρα, όταν η τρίτη τσαγιέρα τοποθετείται, πρέπει να τοποθετηθεί και η τέταρτη και ούτω καθεξής.

Μόνο η επιλογή C έχει μονό αριθμό τσαγιερών, οπότε είναι λανθασμένη. Φυσικά, χρειάζεται να ελέγξετε προσεκτικά τις επιλογές A, B και D για να βεβαιωθείτε ότι το σερβίτσιο είναι τοποθετημένο σύμφωνα με το δεδομένο διάγραμμα.

## Σύνδεση με την Υπολογιστική Σκέψη

Η επιστήμη της πληροφορικής



Αυτό είναι ένα κλασικό πρόβλημα της επιστήμης των υπολογιστών, όπου σας δίνεται ένα σχέδιο (σε αυτή την περίπτωση ένα λεγόμενο αυτόματο) και πρέπει να κατανοήσετε πώς λειτουργεί και ποια αποτελέσματα μπορούν να προκύψουν. Ο πιο εύκολος τρόπος για να το κάνετε αυτό είναι να κοιτάξετε μερικά παραδείγματα – τότε μπορείτε να δείτε τα πρότυπα. Έτσι, όταν ψάχνουμε για μια παραλλαγή στην οποία το σερβίτσιο δεν είναι τοποθετημένο σωστά σύμφωνα με το σχέδιο, θα εστιάσουμε σε ένα από τα πιο σημαντικά μέρη του σχεδίου – την επανάληψη (τοποθέτηση των τσαγιερών). Πρέπει να διασφαλίσουμε ότι ο αριθμός των τσαγιερών είναι πάντα ζυγός στη σειρά των πιάτων.

### Υπολογιστική σκέψη

Σε αυτό το πρόβλημα, όταν ελέγχουμε την ορθότητα των επιλογών (τη σειρά των πιάτων) σύμφωνα με το δεδομένο διάγραμμα, πρέπει να παρατηρήσουμε την αναγνώριση προτύπων, που αποτελεί ένα από τα σημαντικά συστατικά της υπολογιστικής σκέψης. Η αναγνώριση προτύπων, τάσεων και κανονικοτήτων μέσα στο πρόβλημα ή στη λύση του βοηθά στην καλύτερη κατανόηση του προβλήματος και στην εύρεση πιο αποδοτικών λύσεων.

## Χώρα προέλευσης θέματος

Λιθουανία

## 2024-PL-03 – Σχεδιάζοντας ένα ιστιοφόρο

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Σχεδιάζοντας ένα ιστιοφόρο

Η Σοφία θέλει να σχεδιάσει το παρακάτω ιστιοφόρο.

Σκέφτεται τους δύο παρακάτω κανόνες:

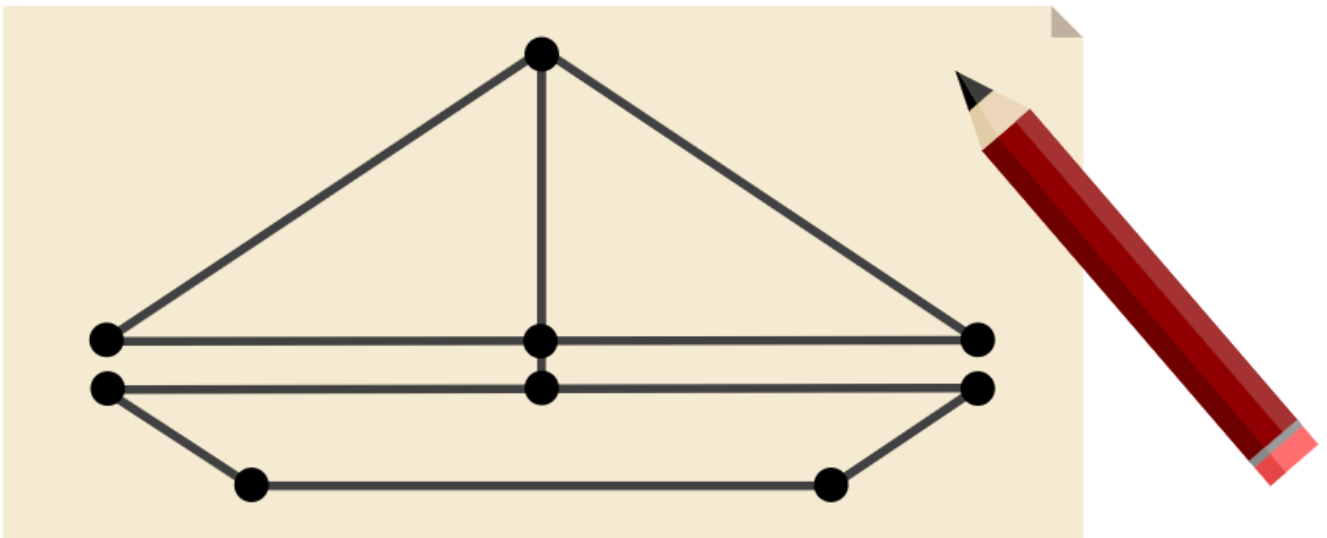
Το ιστιοφόρο πρέπει να σχεδιαστεί με μια γραμμή συνεχόμενη. Η Σοφία δεν πρέπει να σηκώσει το μολύβι από το χαρτί.

Μια γραμμή μπορεί να σχεδιαστεί μόνο μια φορά. Δεν μπορεί να ξανασχεδιαστεί πολλές φορές.

### Ερώτηση

Από πού πρέπει να αρχίσει η Σοφία τη σχεδίαση; Πάτησε στο σημείο της αρχής.

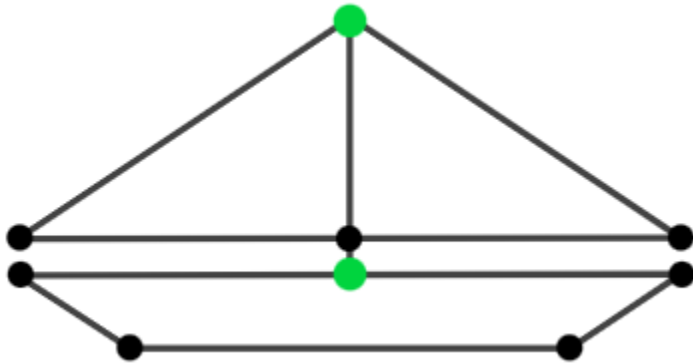
*Πάτησε "Αποθήκευση" όταν τελειώσεις.*



## Απάντηση

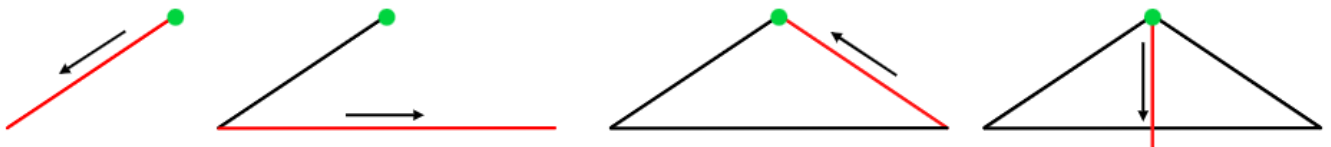
Η Σοφία μπορεί να βρει ποιά σημεία είναι κατάλληλα ως σημεία εκκίνησης, σχεδιάζοντας τις γραμμές από κάθε έναν από τους εννέα πράσινους κύκλους.

Υπάρχουν δύο επιτρεπτά σημεία εκκίνησης.



## Εξήγηση απάντησης

Και τα δύο αυτά σημεία έχουν έναν περιττό αριθμό γραμμών που εξέρχονται από αυτά. Όταν ξεκινάμε από ένα σημείο που έχει περιττό αριθμό εξερχόμενων γραμμών και σχεδιάζουμε την πρώτη γραμμή, πρέπει να επιστρέψουμε στο ίδιο σημείο για να σχεδιάσουμε τη δεύτερη γραμμή. Στη συνέχεια, χρησιμοποιούμε την τρίτη γραμμή για να αλλάξουμε κατεύθυνση ώστε να εξέλθουμε ξανά από το σημείο και να ολοκληρώσουμε το σχέδιο χωρίς να περάσουμε από τις προηγούμενες δύο γραμμές.



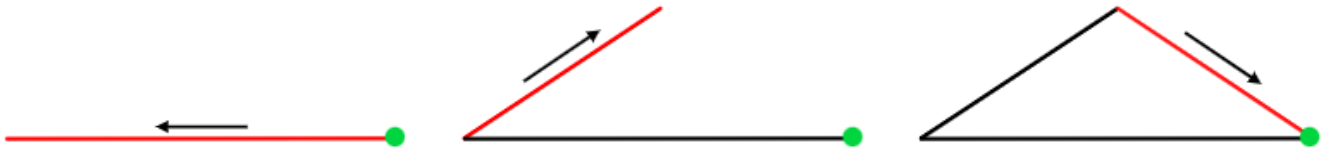
Όλες οι λανθασμένες απαντήσεις είναι σημεία με ζυγό αριθμό γραμμών που εξέρχονται από αυτά. Διότι κάθε φορά που βγαίνουμε από ένα σημείο και επιστρέφουμε σε αυτό, έχουμε σχεδιάσει δύο γραμμές από αυτό (ζυγός αριθμός), και δεν είναι δυνατό να εξέλθουμε ξανά από αυτό χωρίς μια επιπλέον γραμμή (περιττός αριθμός γραμμών).

## Σύνδεση με την Υπολογιστική Σκέψη

## Επιστήμη της πληροφορικής

Το σχέδιο του ιστιοφόρου αποτελείται από σημεία και γραμμές. Στην επιστήμη των υπολογιστών, αυτό ονομάζεται γράφημα. Ένα γράφημα αποτελείται από κόμβους και ακμές. Οι κόμβοι είναι τα σημεία και οι ακμές είναι οι γραμμές που συνδέουν δύο κόμβους.

Στο πρόβλημα αυτό, υπάρχει η ανάγκη να βρεθεί μια διαδρομή που περνά από όλους τους κόμβους και στην οποία όλες οι ακμές εμφανίζονται ακριβώς μία φορά. Οι κόμβοι μπορούν να χρησιμοποιηθούν όσες φορές θέλετε. Για να μπορεί να λυθεί το πρόβλημα, δεν πρέπει να υπάρχουν κόμβοι που να μην μπορούν να χρησιμοποιηθούν καθόλου. Ένα γράφημα που πληροί αυτή την προϋπόθεση λέγεται 'συνδεδεμένο'. Το «Σπίτι του Αγίου Βασίλη» είναι ένας γνωστός γρίφος στον οποίο παιδιά και ενήλικες πρέπει να βρουν μια διαδρομή σε ένα συνδεδεμένο γράφημα.



## Χώρα προέλευσης θέματος

Πολωνία

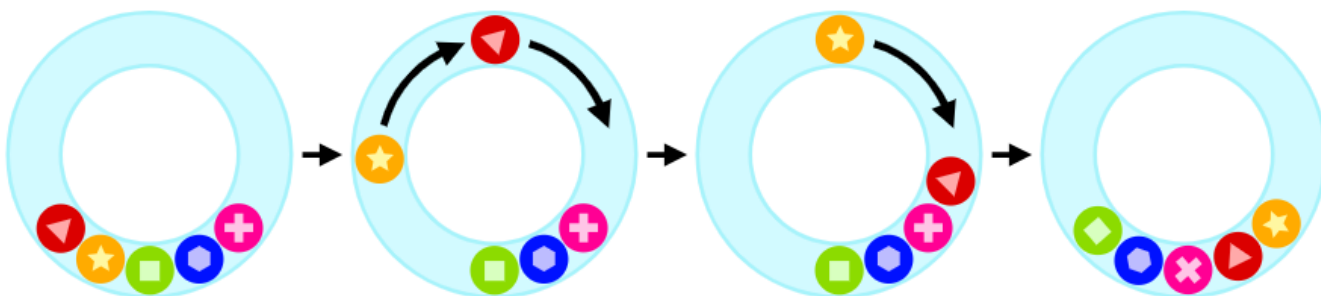
## 2024-SK-01b – Η κουδουνίστρα του Άλκη

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Η κουδουνίστρα του Άλκη

Ο Άλκης έχει μια διαφανή κουδουνίστρα με πολύχρωμες μπάλες. Όταν την κουνάει, μερικές μπάλες περνούν μέσα από την κουδουνίστρα και μετακινούνται στην άλλη πλευρά, όπως μπορείτε να δείτε στην εικόνα.



Ο Άλκης κουνάει την κουδουνίστρα άλλη μια φορά.

### Ερώτηση

Πώς μοιάζει η κουδουνίστρα του Άλκη αφού την κουνήσει;

Σύρετε τις μπάλες στη σωστή θέση μέσα στην κουδουνίστρα.

*Κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.*



## Απάντηση



Όταν οι μπάλες μπουν μέσα στην κουδουνίστρα, δεν μπορείς να αλλάξεις τη σειρά τους. Άρα χρειάζεται μόνο να κοιτάξεις τη σειρά όλων των μπαλών. Μετά την κόκκινη μπάλα



έχει σειρά η κίτρινη



. Μετά την πράσινη μπάλα



είναι η σειρά της μπλε



. Στη συνέχεια παίρνει σειρά η μοβ



και μετά η κόκκινη ξανά



## Εξήγηση απάντησης

Μετά την πράσινη μπάλα, μια μπάλα λείπει, μεταξύ της πράσινης και της μοβ. Σχετικά με την προηγούμενη παράγραφο αυτή πρέπει να είναι η μπλε μπάλα.

Μπορούμε να δούμε ότι μεταξύ της μοβ και της πράσινης μπάλας υπάρχουν δυο μπάλες που λείπουν. Αυτό μας αφήνει με τις επιλογές μιας κίτρινης και μιας κόκκινης μπάλας. Στη σειρά στην αρχή, παρατηρούμε ότι πριν την πράσινη μπάλα πρέπει να υπάρχει μια κίτρινη. Και μεταξύ της μοβ και της κίτρινης μπάλας πρέπει να υπάρχει μια κόκκινη μπάλα.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Οι επιστήμονες υπολογιστών αναπαριστούν και αποθηκεύουν τα δεδομένα με διάφορους τρόπους. Όλες οι αναπαραστάσεις δεδομένων έχουν πλεονεκτήματα που εξαρτώνται από τις ιδιότητες των αντίστοιχων δεδομένων.

Σε αυτό το πρόβλημα χρησιμοποιείται η λεγόμενη κυκλική συνδεδεμένη λίστα, η οποία ονομάζεται "κουδουνίστρα" και τα αποθηκευμένα στοιχεία είναι οι μπάλες. Σε αυτή την κυκλική-συνδεδεμένη λίστα, κάθε στοιχείο έχει ένα λεγόμενο δείκτη προς το επόμενο στοιχείο, δηλαδή δείχνει πιο είναι το επόμενο στοιχείο. Επιπλέον, ο δείκτης του τελευταίου στοιχείου δείχνει στο πρώτο στοιχείο και έτσι κλείνει τον κύκλο.

Αυτό σημαίνει ότι, σε αυτό το πρόβλημα, γνωρίζεις για κάθε μπάλα ποια μπάλα είναι η επόμενη. Μετά την τελευταία μπάλα έρχεται η πρώτη ξανά. Όταν ελέγχεις τη σειρά των μπαλών, μπορείς να ξεκινήσεις από οποιαδήποτε μπάλα και πρέπει να διασχίσεις όλες (ολόκληρη τη λίστα) με τη σωστή σειρά.

Υπάρχουν δύο πλεονεκτήματα όταν χρησιμοποιείς μια κυκλική συνδεδεμένη λίστα για την αποθήκευση στοιχείων. Το πρώτο πλεονέκτημα είναι ότι σε μια κυκλική συνδεδεμένη λίστα δεν υπάρχει τελικό σημείο. Η λίστα επαναλαμβάνεται πάλι από την αρχή, και έτσι τα στοιχεία ποτέ δεν τελειώνουν. Το δεύτερο πλεονέκτημα είναι η αποδοτικότητα της πλοήγησης μέσα στη λίστα. Δηλαδή είναι εύκολο να διατρέξεις τη λίστα χωρίς να χρειάζεται να ακολουθήσεις ειδικούς κανόνες για το τελικό σημείο. Απλώς συνεχίζεις προς μία κατεύθυνση.

Υπολογιστική σκέψη

Σε αυτό το πρόβλημα, προσπάθησες να αναζητήσεις ένα μοτίβο. Αυτό ονομάζεται αναγνώριση μοτίβου. Η αναγνώριση μοτίβων είναι χρήσιμη για να καθορίσεις αν κάποια προβλήματα είναι παρόμοια ή ακόμη και τα ίδια. Τότε, δεν χρειάζεται να λύσεις το καθένα ξεχωριστά, αντίθετα μπορείς να χρησιμοποιήσεις τη λύση σε ένα από αυτά για να σε βοηθήσει να λύσεις το άλλο.

## Χώρα προέλευσης θέματος

Σλοβακία

## 2024-CH-03b – Κάρτες

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού




Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Κάρτες

Η Σταυρούλα συλλέγει κάρτες που δείχνουν τέρατα που ονομάζονται «Ρίκα». Όλες οι κάρτες Ρίκα έχουν χαρακτηριστικά, όπως ένα όνομα ή τον αριθμό των ματιών. Κάθε Ρίκα είναι διαφορετική, οπότε οι τιμές των χαρακτηριστικών μερικές φορές διαφέρουν. Ακολουθούν τρία παραδείγματα καρτών που δείχνουν μερικά από αυτά τα χαρακτηριστικά και τις τιμές τους:



Κάθε τιμή ενός χαρακτηριστικού σημειώνεται με συγκεκριμένο τρόπο:

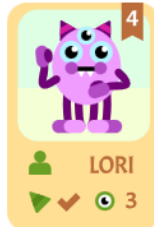
Χαρακτηριστικό	Τύπος τιμής
	Οι τιμές δίνονται ως <b>κείμενο</b>
	Οι τιμές δίνονται ως <b>✘</b> ή <b>✔</b>
	Οι τιμές δίνονται ως <b>αριθμοί</b>



## Ερώτηση

Σε ποια κάρτα σημειώνονται όλες οι τιμές των χαρακτηριστικών στο σωστό είδος;










## Απάντηση




Η σωστή απάντηση είναι :


## Εξήγηση απάντησης

Ο παρακάτω πίνακας , εξηγεί την απάντηση:

	Values on the card			
Attribute	A)	B)	C)	D)
	LORI	LORI	LORI	
				
		3	3	3

Μόνο στην κάρτα B), οι τιμές κάθε χαρακτηριστικού σημειώνεται ότι είναι του σωστού είδους.

- A), Γ) και Δ): Η Ricca με το όνομα Lori έχει 2 κέρατα στην εικόνα, αλλά η τιμή για τα κέρατα,  , μπορεί να είναι μόνο X ή ✓ .

- Απάντηση A): Η Ricca με το όνομα Lori έχει μάτια στην εικόνα, αλλά η τιμή για τα μάτια,  , μπορεί να είναι μόνο ένας αριθμός που σε αυτή την περίπτωση πρέπει να είναι ο αριθμός 3.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Οι τιμές στις κάρτες περιγράφουν τα Riccas. Οι τιμές για τα Riccas θα μπορούσαν να αποθηκευτούν ως μεταβλητές για κάθε κάρτα. Κάθε μεταβλητή έχει έναν τύπο δεδομένων που σχετίζεται με αυτήν και χρησιμοποιείται για να καθορίσει ποιοι τύποι τιμών είναι αποδεκτοί ή όχι. Στις κάρτες Ricca που παρουσιάζονται, το όνομα αποθηκεύεται ως κείμενο, τα μάτια αποθηκεύονται ως αριθμός και τα κέρατα είναι είτε ναι είτε όχι.

Ανάλογα με τον τρόπο που αποθηκεύουμε τα δεδομένα στον υπολογιστή, αυτός μπορεί να εκτελέσει διαφορετικούς υπολογισμούς (ή πράξεις). Πριν αποθηκεύσουμε δεδομένα στον υπολογιστή, πρέπει να σκεφτούμε τι θέλουμε να κάνουμε με αυτά αργότερα. Τότε μπορούμε να αποφασίσουμε ποιον τύπο δεδομένων θα χρησιμοποιήσουμε. Ο τύπος δεδομένων όχι μόνο καθορίζει ποιες τιμές μπορούν να γίνουν αποδεκτές, αλλά υποδεικνύει επίσης ποιες πράξεις μπορούν να εκτελεστούν.

### Υπολογιστική σκέψη

Για την επίλυση αυτής της εργασίας απαιτείται αφαίρεση: Αρχικά, οι συμμετέχοντες πρέπει να φιλτράρουν τις σχετικές πληροφορίες και να επικυρώσουν τις δεδομένες τιμές. Το δεύτερο βήμα είναι να διακρίνουν μεταξύ μιας σωστής τιμής και μιας σωστής τιμής στον σωστό τύπο δεδομένων.

## Χώρα προέλευσης θέματος

Ελβετία









## 2024-KR-02 - Πάρτυ στην αυλή

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Πάρτυ στην αυλή

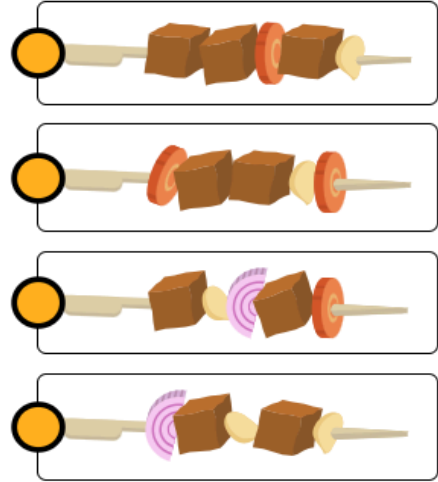
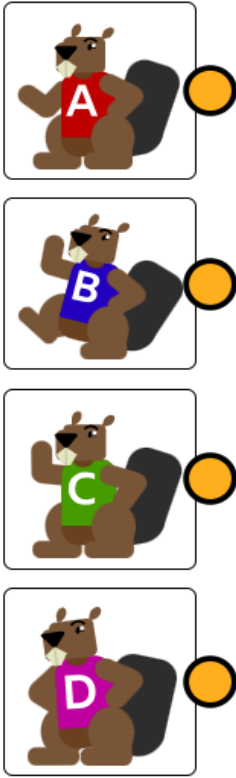
Στο πάρτι στην αυλή, τέσσερεις κάστορες περιγράφουν πως θέλουν να είναι το σουβλάκι τους.

			
Χρειάζομαι 	Δεν τρώω 	Όσο περισσότερο  γίνεται	Όσο περισσότερο  γίνεται

### Ερώτηση

Υπάρχουν τέσσερα σουβλάκια. Μοίρασε τα ώστε όλοι οι κάστορες να είναι όσο δυνατόν ικανοποιημένοι.

*Σχεδίασε μια γραμμή μεταξύ των καστόρων και του κάθε ενός σουβλακιού για να τα ενώσεις. Πάτησε "Αποθήκευση" όταν τελειώσεις.*



### Απάντηση

Η σωστή απάντηση είναι:


### Εξήγηση απάντησης

Ας αριθμήσουμε τα σουβλάκια:



Μόνο το 4ο σουβλάκι έχει παραπάνω από ένα κομμάτι σκόρδο. Ο κάστορας με το γράμμα C παίρνει το 4ο σουβλάκι.

Μόνο το 1ο σουβλάκι έχει περισσότερα από δύο κομμάτια κρέας. Ο κάστορας με το γράμμα D παίρνει το 1ο σουβλάκι.

Από τα δύο σουβλάκια που απομένουν, μόνο το 3ο έχει κρεμμύδι, αυτό λοιπόν πρέπει να το πάρει ο κάστορας με το γράμμα A και όχι με το γράμμα B.

Ο κάστορας με το γράμμα B παίρνει το 2ο σουβλάκι.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Το Πρόβλημα Ικανοποίησης Περιορισμών (Constraint Satisfaction Problem - CSP) είναι μία από τις βασικές έννοιες στην επιστήμη των υπολογιστών. Ένα CSP είναι το πρόβλημα εύρεσης μιας τιμής που ικανοποιεί ορισμένους περιορισμούς σε ένα σύνολο μεταβλητών. Αυτά τα προβλήματα συναντώνται σε διάφορους τομείς, όπως ο προγραμματισμός μιας εργασίας, ο σχεδιασμός, η κατανομή πόρων, ο σχεδιασμός μιας διάταξης και η υπολογιστική όραση. Στην πληροφορική, το CSP χρησιμοποιείται ως μια σημαντική προσέγγιση για την επίλυση προβλημάτων. Η πληροφορική περιλαμβάνει την ανάλυση δεδομένων, την ανάπτυξη αλγορίθμων και τον σχεδιασμό λογισμικού και συστημάτων, όπου το CSP μπορεί να αξιοποιηθεί για την ανάπτυξη αποδοτικών λύσεων. Για παράδειγμα, το CSP μπορεί να χρησιμοποιηθεί σε συστήματα βάσεων δεδομένων όταν χρειάζεται να οριστούν περιορισμοί για τη διατήρηση της συνέπειας των δεδομένων και να βρεθούν λύσεις που να τους ικανοποιούν.

### Υπολογιστική Σκέψη

Αυτό το πρόβλημα είναι ένα πρόβλημα ικανοποίησης περιορισμών, που σημαίνει ότι για να το λύσετε, πρέπει να οργανώσετε τις συνθήκες που θέλει κάθε κάστορας, εφαρμόζοντας λογική σκέψη. Στη συνέχεια, πρέπει να λύσουμε τις επιλογές που περιορίζονται από τους περιορισμούς. Υποθέτοντας ότι μπορείτε να ικανοποιήσετε όλους τους κάστορες, ξεκινάτε ικανοποιώντας πρώτα εκείνον που έχει τις πιο αυστηρές συνθήκες. Έπειτα, ικανοποιείτε την επόμενη πιο αυστηρή συνθήκη, και συνεχίζετε έτσι μέχρι να ικανοποιηθούν όλες οι συνθήκες. Αυτή η διαδικασία επιτρέπει στους μαθητές να βιώσουν τη διαδικασία διάσπασης ενός σύνθετου προβλήματος σε μικρότερα υπο-προβλήματα, να ορίσουν περιορισμούς για κάθε υπο-πρόβλημα και στη συνέχεια να βρουν μια λύση που να ικανοποιεί αυτούς τους περιορισμούς.

## Χώρα προέλευσης θέματος

Νότια Κορέα

## 2024-PL-04 – Η διαδρομή του ρομπότ

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Η διαδρομή του ρομπότ

Το ρομπότ του Βασίλη κινείται σε πλακίδια ακολουθώντας ένα σύνολο κανόνων:

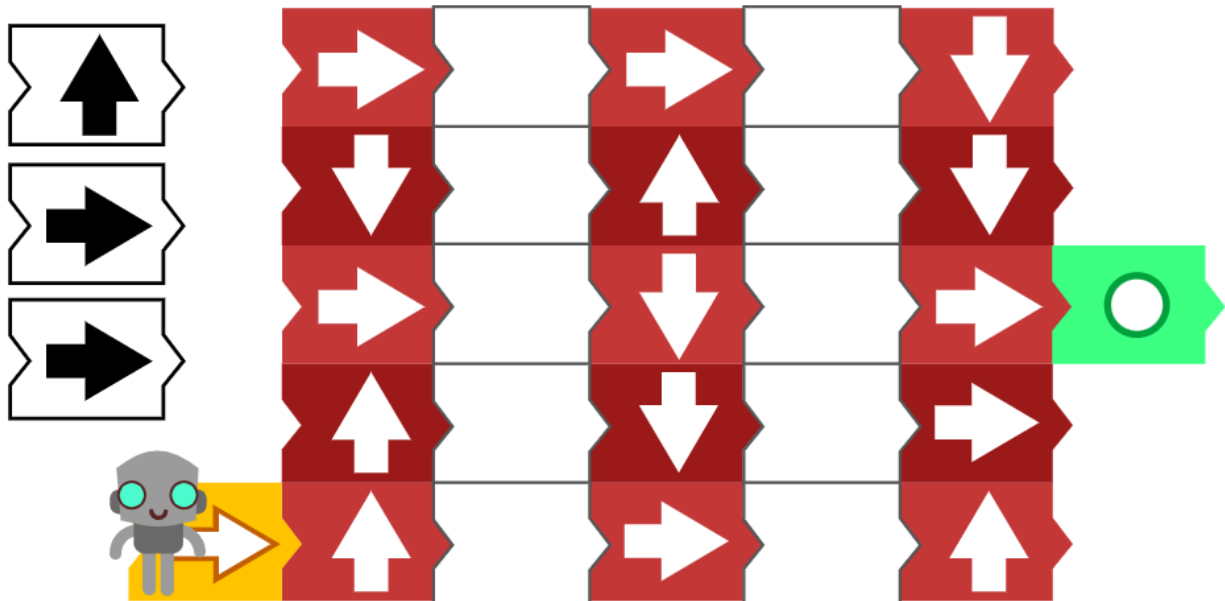
- Το ρομπότ ξεκινά από το πιο αριστερό πλακίδιο, όπως βλέπετε στην παρακάτω εικόνα.
- Το ρομπότ κινείται προς την κατεύθυνση του βέλους στο οποίο στέκεται.
- Εάν το ρομπότ απομακρυνθεί από τα πλακίδια, σταματά.

Ορισμένα πλακίδια είναι προκαθορισμένα όπως φαίνεται παρακάτω. Στον Βασίλη δίνονται τρία επιπλέον πλακίδια.

### Ερώτηση

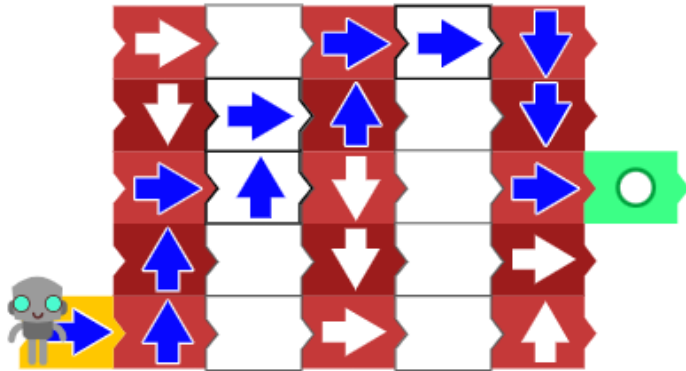
Βοηθήστε τον Βασίλη να τοποθετήσει και τα τρία επιπλέον πλακίδια στις κενές περιοχές, έτσι ώστε το ρομπότ του να φτάσει στο πράσινο πλακίδιο με τον κύκλο.

*Σύρτε τα βέλη στη σωστή θέση. Κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.*



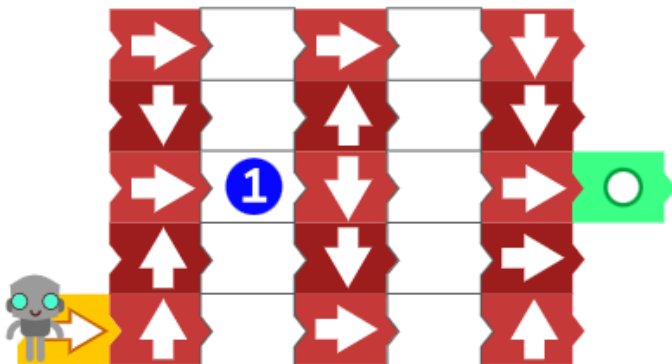
## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Όπως έχει ήδη υποδειχθεί, το ρομπότ του Βασίλη θα κινηθεί πρώτα προς τα δεξιά, δύο φορές προς τα πάνω και πάλι προς τα δεξιά. Αυτό σημαίνει ότι μια κάρτα πρέπει να τοποθετηθεί στη θέση που σημειώνεται με το 1:



Οι δύο τύποι διαθέσιμων καρτών είναι «πάνω» ή «δεξιά». Αν επιλέξετε «δεξιά», το ρομπότ θα κινηθεί προς τα δεξιά, μετά δύο φορές προς τα κάτω και πάλι προς τα δεξιά. Τώρα θα μπορούσατε είτε να επιλέξετε πάλι «δεξιά» είτε «πάνω» και μετά «δεξιά». Και στις δύο περιπτώσεις, το ρομπότ θα προσγειωνόταν κάτω από τον επιθυμητό στόχο:





Κατά κανόνα, θα λύσετε αυτό το πρόβλημα όπως περιγράφεται στην επεξήγηση της απάντησης: Θα δοκιμάσετε έναν τρόπο και αν αυτός δεν οδηγεί στο στόχο, θα επιστρέψετε και θα δοκιμάσετε έναν άλλο τρόπο.

Στην επιστήμη των υπολογιστών, αυτό ονομάζεται αναζήτηση σε βάθος (depth-first search), επειδή πρώτα δοκιμάζεται εντελώς ένας δρόμος και μετά δοκιμάζεται ο επόμενος. Επειδή συχνά πρέπει να πάτε ένα ή περισσότερα βήματα πίσω για να δοκιμάσετε μια διαφορετική λύση, κάνετε backtracking. Σε κάθε περίπτωση, όμως, λύνετε το πρόβλημα με την προς τα εμπρός ανίχνευση.

Ωστόσο, ορισμένα προβλήματα επιλύονται καλύτερα προς τα πίσω, κοιτάζοντας από τον στόχο για να δείτε πώς θα μπορούσατε να φτάσετε εκεί. Στην πραγματικότητα, αυτό το είδος προβλήματος δεν είναι κατάλληλο για να λυθεί προς τα πίσω, διότι ενώ όταν πηγαίνετε προς τα εμπρός ξέρετε πάντα ποια κάρτα είναι η επόμενη, όταν πηγαίνετε προς τα πίσω μπορεί να σημαίνει ότι έρχεστε από πολλές κάρτες (το βέλος δείχνει προς τα εμπρός και πηγαίνοντας προς τα πίσω πρέπει να ψάξετε όλες τις κάρτες γύρω του για να δείτε αν δείχνουν προς την κατεύθυνση της κάρτας).

Αλλά στη συγκεκριμένη περίπτωση, εξακολουθεί να είναι εύκολο να δείτε ποια λύση μπορεί να είναι η μόνη σωστή.

### Υπολογιστική σκέψη

Ένας υπολογιστής είναι καλός στο να δοκιμάζει απλώς όλες τις πιθανές περιπτώσεις. Δεδομένου ότι υπάρχουν μόνο (για έναν υπολογιστή) πολύ λίγες πιθανές θέσεις και πολύ λίγες πιθανές κάρτες διαθέσιμες, ένας υπολογιστής μπορεί πολύ γρήγορα να δοκιμάσει απλά όλες τις πιθανότητες. Ένας τέτοιος σχετικά «αφελής» αλγόριθμος θα έλυne καλά το πρόβλημα.

Ωστόσο, οι άνθρωποι χρειάζονται λίγο περισσότερο χρόνο για να δοκιμάσουν τα πάντα. Επομένως, αξίζει να αναλύσουμε το πρόβλημα πιο προσεκτικά και να διατυπώσουμε περιορισμούς που μειώνουν κατά πολύ τον αριθμό των περιπτώσεων που πρέπει να δοκιμαστούν. Για παράδειγμα, μια κάρτα πρέπει να βρίσκεται στην τρίτη υψηλότερη θέση της αριστερής κόκκινης στήλης. Αυτό περιορίζει σημαντικά τον χώρο αναζήτησης πιθανών λύσεων, έτσι ώστε τελικά να χρειάζεται να δοκιμαστούν μόνο δύο παραλλαγές, όπως φαίνεται στην επεξήγηση της απάντησης.

Η υπολογιστική σκέψη δεν σημαίνει ότι απλώς δοκιμάζουμε όλες τις πιθανές περιπτώσεις όπως κάνει ένας υπολογιστής, αλλά ότι λαμβάνουμε υπόψη απλουστεύσεις που χρησιμοποιούν και οι επιστήμονες της πληροφορικής. Σε αυτή την περίπτωση, αυτό θα μπορούσε να είναι μια εφαρμοσμένη αναζήτηση σε βάθος με οπισθοδρόμηση ή μια αναζήτηση προς τα πίσω αντί για αναζήτηση προς τα εμπρός.

## Χώρα προέλευσης θέματος

Πολωνία

## 2024-TW-03-UK – Το πιο μακρύ βραχιόλι

Ηλικιακή ομάδα: Γ' - Δ' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Το πιο μακρύ βραχιόλι

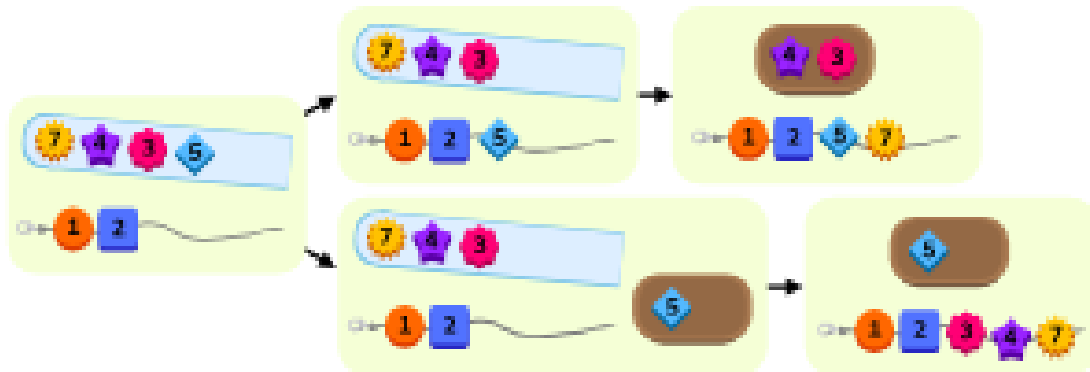
Ο Μάριος φτιάχνει ένα βραχιόλι. Παίρνει χάντρες με αριθμούς από ένα γυάλινο σωλήνα. Κολλάει μερικές χάντρες από αυτές και βάζει τις άλλες στο καφέ κουτί και δεν τις χρησιμοποιεί. Επιτρέπεται να περάσει μια χάντρα μόνο αν:

το κορδόνι είναι άδειο, ή

η χάντρα έχει μεγαλύτερο αριθμό από την τελευταία χάντρα στο κορδόνι.

Σε αυτό το παράδειγμα, η τελευταία χάντρα στο κορδόνι είναι η χάντρα 2. Στη συνέχεια, ο Μάριος μπορεί να περάσει τη χάντρα 5 από τον σωλήνα. Μπορεί επίσης να την αφήσει στο καφέ κουτί και να μην τη χρησιμοποιήσει.

Αν περάσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με τέσσερις χάντρες 1257. Αν δεν χρησιμοποιήσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με περισσότερες χάντρες 12347.



Ο Μάριος φτιάχνει ένα καινούργιο βραχιόλι από τις χάντρες αυτού του σωλήνα:



## Ερώτηση

Ποιος είναι ο μέγιστος αριθμός χαντρών που μπορεί να περάσει από το κορδόνι;

## Απάντηση

6

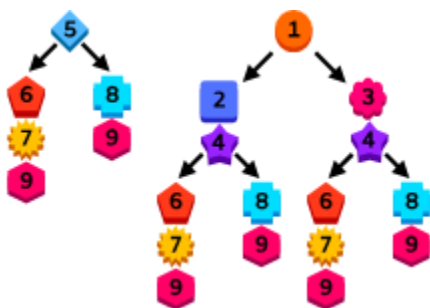
## Εξήγηση απάντησης

Η απάντηση είναι 6.

Θα μπορούσαμε να απαριθμήσουμε όλα τα βραχιόλια για αυτό το σωληνάριο και να δούμε ποια έχουν τον μεγαλύτερο αριθμό χαντρών. Αλλά αυτό θα ήταν χρονοβόρο. Ας ρίξουμε μια πιο προσεκτική ματιά στους αριθμούς:



Η πρώτη χάντρα είναι ο αριθμός 5. Όταν αυτή περνάει στο κορδόνι, οι επόμενες δυνατές χάντρες είναι μόνο οι 6, 7, 8 και 9, επομένως οι πιθανές ακολουθίες θα ήταν 5679 και 589. Εάν η χάντρα 5 αφηθεί στην άκρη και η επόμενη χάντρα, η χάντρα 1, μπει στο κορδόνι, υπάρχουν περισσότερες πιθανές ακολουθίες. Αυτό μπορεί να αναπαρασταθεί ξεκάθαρα σε ένα δέντρο αποφάσεων:



Εάν η χάντρα 1 αφηθεί επίσης στην άκρη και μία διαφορετική χάντρα περαστεί πρώτα, οι πιθανές αλληλουχίες περιέχονται ήδη σε μία από τις παραπάνω ακολουθίες. Εάν ξεκινήσετε με την χάντρα 2, για παράδειγμα, θα λάβετε 24679 ή 2489, που περιέχονται στις μεγαλύτερες ακολουθίες 124679 και 12489 αντίστοιχα.

Τα βραχιόλια με τον μεγαλύτερο αριθμό χαντρών ξεκινούν με τη χάντρα 1 και αποτελούνται από 6 χάντρες: 124679 ή 123679.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Κάθε σωλήνας περιέχει μια ακολουθία αριθμημένων χαντρών. Εάν ο Μάριος ακολουθήσει τον δεύτερο κανόνα (μόνο οι χάντρες στο κορδόνι που έχουν μεγαλύτερο αριθμό από την τελευταία), θα πάρει ένα βραχιόλι με αύξουσα ακολουθία. Για να κατασκευαστεί ένα βραχιόλι με όσο το δυνατόν περισσότερες χάντρες, πρέπει να καθορίσει η μεγαλύτερη ανοδική ακολουθία.

Για μεγάλες ακολουθίες, θα χρειαζόταν πολύς χρόνος για να σχηματιστούν όλες οι πιθανές αύξουσες υποακολουθίες και στη συνέχεια να προσδιοριστεί η μεγαλύτερη. Εάν ο σωλήνας περιείχε 20 χάντρες, για παράδειγμα, η υπολογιστική προσπάθεια θα ήταν της τάξης ενός εκατομμυρίου βημάτων.

Ευτυχώς, στην επιστήμη των υπολογιστών έχουν αναπτυχθεί αλγόριθμοι που μπορούν να βρουν τη μεγαλύτερη αύξουσα υποακολουθία πιο γρήγορα. Αυτό γίνεται χρησιμοποιώντας μια τεχνική που ονομάζεται δυναμικός προγραμματισμός. Η τάξη μεγέθους της υπολογιστικής προσπάθειας για αυτούς τους γρήγορους αλγόριθμους είναι μικρότερη από εκατό βήματα για έναν σωλήνα με 20 χάντρες.

## Χώρα προέλευσης θέματος



Ταϊβάν

## 2024-BG-01b - Μπάλες

Ηλικιακή ομάδα: Γ' – Δ' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Μπάλες

Παρακάτω βλέπετε μια ακολουθία από κόκκινες  και μπλε  μπάλες:



Για κάθε μπάλα, μετράμε τώρα πόσες μπλε μπάλες βρίσκονται στα δεξιά της (συμπεριλαμβανομένης της ίδιας της μπάλας) και το σημειώνουμε ως εξής:

3, 3, 2, 1, 1, 1.

Στο επόμενο βήμα, σημειώνουμε ξανά την ίδια ακολουθία. Αλλά αν ο αριθμός είναι ζυγός (άρτιος), σημειώνουμε ένα 0, αν είναι μονός (περιττός), τότε ένα 1. Αυτό οδηγεί σε αυτή την ακολουθία:

110111

### Ερώτηση

Δίνεται η παρακάτω ακολουθία των αριθμών 0 και 1:

01110100

Κατασκευάστε μια ακολουθία μπλε και κόκκινων μπάλων που να ταιριάζει με αυτή την αναπαράσταση.



## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Υπάρχουν διάφοροι τρόποι για να βρείτε αυτή την απάντηση. Ένας τρόπος είναι να περάσετε την ακολουθία των αριθμών από τα δεξιά προς τα αριστερά και να συμπεράνετε λογικά αν η τρέχουσα μπάλα πρέπει να είναι κόκκινη ή μπλε.

Ας δούμε τις μπάλες με τη σειρά:

1. Ας κοιτάξουμε την πρώτη μπάλα στα δεξιά. Παίρνουμε το 0 σε αυτό το σημείο ως ένδειξη ότι η μπάλα πρέπει να είναι κόκκινη, επειδή μόνο μια κόκκινη μπάλα διατηρεί το άθροισμα όλων των μπλε μπαλών ίσο.

0	1	1	1	0	1	0	0
							Κόκκινη

2. Στη δεύτερη θέση από τα δεξιά βρίσκουμε ένα 0. Το επιχείρημα παραμένει το ίδιο όπως και πριν: μόνο με μια κόκκινη μπάλα το άθροισμα όλων των μπλε μπαλών παραμένει σωστό και συμπεριλαμβανομένης αυτής της θέσης αυτή τη στιγμή.

0	1	1	1	0	1	0	0
						Κόκκινη	Κόκκινη

3. Στην τρίτη θέση από τα δεξιά βρίσκουμε ένα 1, το οποίο υποδηλώνει ότι ο αριθμός των μπλε μπαλών θα πρέπει να είναι μονός από αυτό το σημείο και μετά. Εφόσον γνωρίζουμε ήδη ότι οι δύο μπάλες στα δεξιά είναι κόκκινες, άρα η τρέχουσα μπάλα πρέπει να είναι μπλε για να μετακινηθούμε από έναν ευθύγραμμο σε έναν μονό αριθμό μπλε μπαλών.

0	1	1	1	0	1	0	0
					Μπλε	Κόκκινη	Κόκκινη

4. Στην τέταρτη θέση από δεξιά βρίσκουμε ένα 0. Για να αντικαταστήσουμε τις προηγούμενες μονές μπάλες (2x κόκκινες και 1x μπλε) για να πάρουμε έναν ζυγό αριθμό μπλε μπαλών, ο αριθμός των μπλε μπαλών της τέταρτης μπάλας πρέπει να είναι μπλε.

0	1	1	1	0	1	0	0
				Μπλε	Μπλε	Κόκκινη	Κόκκινη

5. Στην πέμπτη θέση από δεξιά βρίσκουμε ένα 1. Αφού το προηγούμενο άθροισμα των μπλε μπαλών ήταν ζυγό και τώρα θα είναι μονό, η νέα μπάλα πρέπει να είναι επίσης μπλε.

0	1	1	1	0	1	0	0
			Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

6. Στην έκτη θέση από δεξιά βρίσκεται ο αριθμός 1. Εφόσον ο προηγούμενος αριθμός μπλε μπαλών ήταν περιττός, η τρέχουσα μπάλα πρέπει να είναι κόκκινη, ώστε ο αριθμός που εμφανίζεται να είναι περιττός.

0	1	1	1	0	1	0	0
		Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη



7. Στην έβδομη θέση από τα δεξιά βρίσκουμε άλλο ένα 1. Η τρέχουσα ακολουθία περιέχει ήδη έναν περιττό αριθμό μπλε μπαλών- για να διατηρηθεί αυτή η κατάσταση, η νέα μπάλα πρέπει να έχει κόκκινο χρώμα.

0	1	1	1	0	1	0	0
	Κόκκινη	Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

8. Στην όγδοη θέση από τα δεξιά (πρώτη θέση από τα αριστερά) βρίσκουμε τον αριθμό 0. Για να γίνει και πάλι ζυγός ο μονός αριθμός μπλε μπαλών, πρέπει να είναι μια μπλε μπάλα.

0	1	1	1	0	1	0	0
Μπλε	Κόκκινη	Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

Συνοπτικά, σε κάθε μετάβαση από το 0 στο 1 ή από το 1 στο 0 πρέπει να χρησιμοποιείται μια μπλε μπάλα, ενώ μια συνεχής ακολουθία των ίδιων αριθμών υποδεικνύει κόκκινες μπάλες. Αυτές δεν επηρεάζουν το άθροισμα των μπλε μπαλών:

0	1	1	1	0	1	0	0
4 Μπλε	3	3	3 Μπλε	2 Μπλε	1 Μπλε	0	0
	Κόκκινες	Κόκκινες				Κόκκινες	Κόκκινες

Τότε η ακολουθία των μπλε μπαλών είναι 43332100.



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

- Η κωδικοποίηση χρησιμοποιείται για την αναπαράσταση πολύπλοκων θεμάτων με τρόπο ώστε οι υπολογιστές να μπορούν εύκολα να τα επεξεργαστούν. Συχνά άσχετα δεδομένα μπορούν να παραλειφθούν χωρίς να επηρεαστούν τα κεντρικά στοιχεία της πληροφορίας. Αυτή η εργασία προτείνει μια κωδικοποίηση του τρόπου με τον οποίο μπορεί να αποθηκευτεί ο ακριβής αριθμός και η θέση των μπλε και κόκκινων μπαλών χωρίς τα ίδια τα πραγματικά χρώματα. Λόγω της επιλεγμένης κωδικοποίησης βρισκόμαστε στη θέση της αρχικής πληροφορίας (τα χρώματα και η θέση των αρχικών μπαλών).
- Υπό ορισμένες συνθήκες, η κωδικοποίηση μπορεί επίσης να εξοικονομήσει αποθηκευτικό χώρο, κάτι που ονομάζουμε συμπίεση. Για την αναπαράσταση δυαδικής πληροφορίας (όπως στην πληροφορία αν ένας αριθμός είναι ζυγός ή περιττός) αρκεί ένα μόνο bit πληροφορίας, ενώ για την αποθήκευση οποιουδήποτε αριθμού ή χρώματος απαιτείται περισσότερος αποθηκευτικός χώρος.
- Ως πρώτο ενδιάμεσο βήμα, προσδιορίστηκε μια αθροιστική συχνότητα για τον υπολογισμό του αθροίσματος των στοιχείων που έχουν ένα συγκεκριμένο χαρακτηριστικό (εδώ συγκεκριμένα τον αριθμό των μπλε μπαλών από την τρέχουσα θέση). Η αθροιστική συχνότητα είναι ένα εργαλείο της στατιστικής, το οποίο χαρακτηρίζεται ιδιαίτερα από το γεγονός ότι η πληροφορία που προκύπτει συσσωρεύεται (εδώ όχι μόνο μια ένδειξη για το χρώμα της τρέχουσας μπάλας, αλλά και για τα χρώματα των μπαλών πίσω από αυτήν). Οι λύσεις προβλημάτων που χρησιμοποιούν τη σωρευτική συχνότητα συχνά επωφελούνται από το γεγονός ότι μπορείτε να επεξεργαστείτε πολλές αιτήσεις γρήγορα χωρίς να χρειάζεται να έχετε πρόσβαση στα αρχικά δεδομένα.

### Υπολογιστική σκέψη

Κατά τη διάρκεια της διαδικασίας κωδικοποίησης ενός προβλήματος, οι μαθητές εξασκούν την ικανότητα της αφαίρεσης: παραλείπονται οι άσχετες πληροφορίες και γίνεται σταδιακή μείωση των λεπτομερειών, χωρίς να χάνεται το κεντρικό πληροφοριακό περιεχόμενο.

## Χώρα προέλευσης θέματος

Βουλγαρία

## 2024-FI-01 - Σελίδα ζωγραφικής

Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

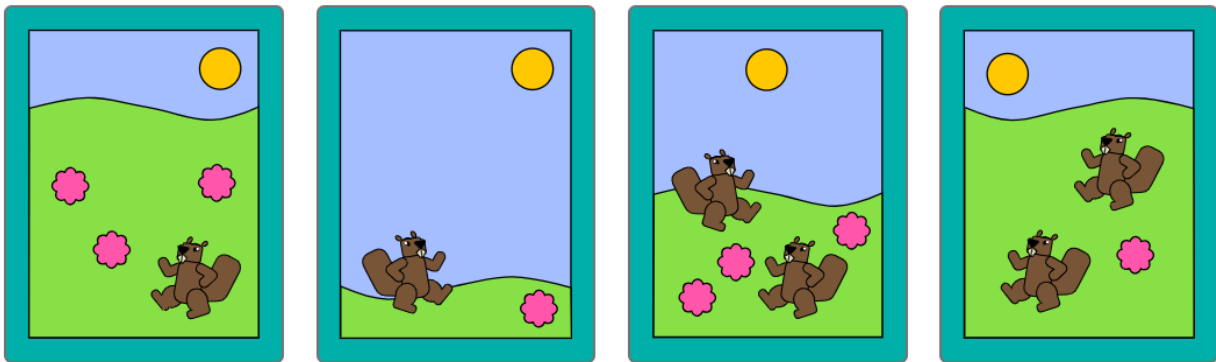
Θέμα : Σελίδα ζωγραφικής

Η Ολυμπία ξεκίνησε με πέντε γεμάτα σωληνάρια χρώματος. Στη ζωγραφιά της, όσο μεγαλύτερη ήταν η συνολική επιφάνεια ενός χρώματος, τόσο περισσότερο χρώμα από εκείνο χρησιμοποίησε.

Όταν η Ολυμπία τελείωσε τη ζωγραφική της, αυτή ήταν η ποσότητα χρώματος που είχε απομείνει σε κάθε σωληνάριο:



Κάστορες Ήλιος Λουλούδια Χορτάρι Ουρανός



Ερώτηση

Ποια ζωγραφιά μπορεί να είναι της Ολυμπίας;

## Απάντηση

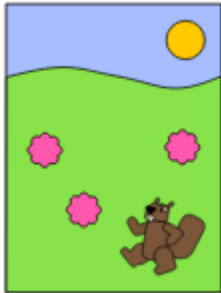
Η σωστή απάντηση είναι το Δ.

## Εξήγηση απάντησης

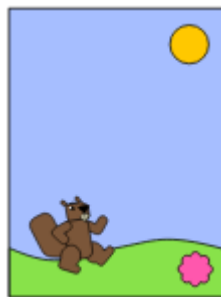
Η σωστή απάντηση είναι η παρακάτω:



Για να βρούμε τη λύση, συγκρίνουμε τις χρωματιστές περιοχές της εικόνας με το περιεχόμενο των σωλήνων βαφής. Μπορείτε να δείτε από τους σωλήνες βαφής ότι το πράσινο χρησιμοποιήθηκε περισσότερο και μετά το μπλε. Αυτό ταιριάζει με εικόνες



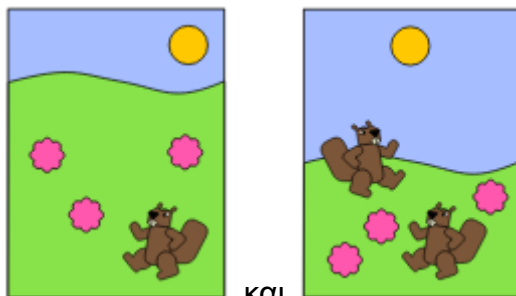
, και . Σε αυτές τις εικόνες, η πράσινη περιοχή είναι η μεγαλύτερη και η μπλε περιοχή είναι η δεύτερη μεγαλύτερη. Μπορούμε



να αποκλείσουμε την εικόνα μεγαλύτερη από την πράσινη.

γιατί η μπλε περιοχή είναι πολύ

Οι σωλήνες βαφής δείχνουν επίσης ότι η Ολυμπία χρησιμοποίησε εξίσου κίτρινο με



βιολετί. Σε εικόνες και , η συνολική βιολετί επιφάνεια (3 λουλούδια) είναι περίπου 3 φορές μεγαλύτερη από την κίτρινη περιοχή (ήλιος). Μόνο



στην εικόνα είναι οι κίτρινες και βιολετί περιοχές (ήλιος και λουλούδι) περίπου στο ίδιο μέγεθος.



Η απάντηση είναι λοιπόν .

Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Σε αυτό το ερώτημα, αναγνωρίσατε μια εικόνα εξετάζοντας τα χρώματα. Κάνετε στον εαυτό σας ερωτήσεις όπως: Ποια χρώματα εμφανίζονται; Ποιο χρώμα χρησιμοποιείται συχνότερα; Ποια χρώματα χρησιμοποιούνται λιγότερο συχνά; Χρησιμοποιείται περισσότερο ένα χρώμα από ένα άλλο; Για να απαντήσετε σε αυτές τις ερωτήσεις, έπρεπε να κάνετε διάκριση μεταξύ διαφορετικών περιοχών στην εικόνα, να υπολογίσετε το μέγεθος των περιοχών, να συγκρίνετε τις τιμές της περιοχής και να λάβετε αποφάσεις. Τέτοιου είδους δοκιμασίες μπορούν επίσης να πραγματοποιηθούν αυτόματα από προγράμματα υπολογιστή. Τα προγράμματα υπολογιστών μπορούν να σας βοηθήσουν να βρείτε φωτογραφίες που είναι αποθηκευμένες στο κινητό σας τηλέφωνο. Εξετάζοντας τα χρώματα, ένα πρόγραμμα υπολογιστή μπορεί να διακρίνει φωτογραφίες τοπίων από φωτογραφίες πορτραίτου καθώς τα τοπία έχουν πολύ μπλε στην κορυφή και πολύ πράσινο στο κάτω μέρος. Αναλύοντας το χρώμα των δορυφορικών φωτογραφιών, οι υπολογιστές μπορούν να προσδιορίσουν πόσο πυκνή βλάστηση έχει μια περιοχή ή αν έχει γίνει παράνομη αποψίλωση των δασών σε ένα φυσικό καταφύγιο. Ο τομέας της επιστήμης των υπολογιστών που εξετάζει αυτόματα τις εικόνες ονομάζεται όραση υπολογιστή.

### Υπολογιστική σκέψη

Ο μαθητής πρέπει να κατανοήσει ότι η αναλογία των χρωμάτων στην εικόνα αντιπροσωπεύεται από την ποσότητα του χρώματος που λείπει από τους σωλήνες. Όταν μελετά την εικόνα, χρησιμοποιεί την ικανότητα να αποσυνθέτει ή να χωρίζει το σύνολο σε μικρότερα κομμάτια, όπου υπάρχουν ποσότητες διαφορετικών χρωμάτων. Η αποσύνθεση (Decomposition) είναι μια από τις βασικές δεξιότητες υπολογιστικής σκέψης.

## Χώρα προέλευσης θέματος

Φινλανδία









## 2024-KR-02-GR - Πάρτυ στην αυλή

Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Πάρτυ στην αυλή

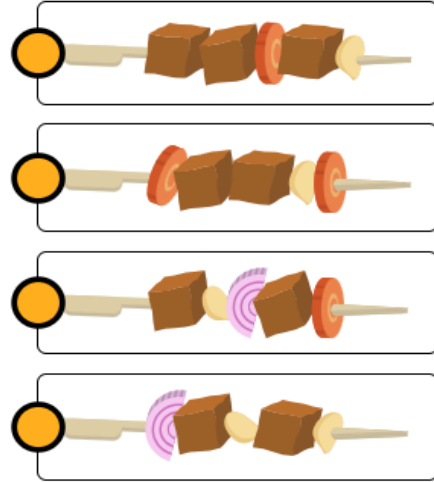
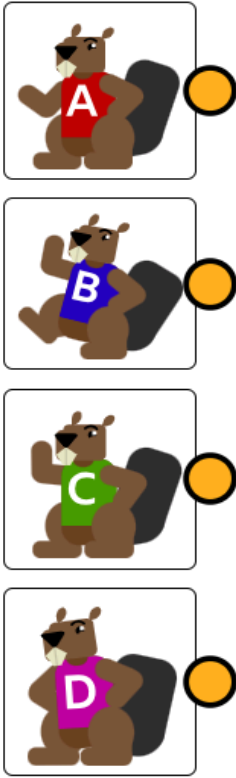
Στο πάρτι στην αυλή, τέσσερις κάστορες περιγράφουν πως θέλουν να είναι το σουβλάκι τους.

			
Χρειάζομαι 	Δεν τρώω 	Όσο περισσότερο  γίνεται	Όσο περισσότερο  γίνεται

### Ερώτηση

Υπάρχουν τέσσερα σουβλάκια. Μοίρασε τα ώστε όλοι οι κάστορες να είναι όσο το δυνατόν ικανοποιημένοι.

Σχεδίασε μια γραμμή μεταξύ των καστόρων και του κάθε ενός σουβλακιού για να τα ενώσεις. Πάτησε "Αποθήκευση" όταν τελειώσεις.



## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Ας αριθμήσουμε τα σουβλάκια:





Μόνο το 4ο σουβλάκι έχει παραπάνω από ένα κομμάτι σκόρδο. Ο κάστορας με το γράμμα C παίρνει το 4ο σουβλάκι.

Μόνο το 1ο σουβλάκι έχει περισσότερα από δύο κομμάτια κρέας. Ο κάστορας με το γράμμα D παίρνει το 1ο σουβλάκι.

Από τα δύο σουβλάκια που απομένουν, μόνο το 3ο έχει κρεμμύδι, αυτό λοιπόν πρέπει να το πάρει ο κάστορας με το γράμμα A και όχι με το γράμμα B.

Ο κάστορας με το γράμμα B παίρνει το 2ο σουβλάκι.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Το Πρόβλημα Ικανοποίησης Περιορισμών (Constraint Satisfaction Problem - CSP) είναι μία από τις βασικές έννοιες στην επιστήμη των υπολογιστών. Ένα CSP είναι το πρόβλημα εύρεσης μιας τιμής που ικανοποιεί ορισμένους περιορισμούς σε ένα σύνολο μεταβλητών. Αυτά τα προβλήματα συναντώνται σε διάφορους τομείς, όπως ο προγραμματισμός μιας εργασίας, ο σχεδιασμός, η κατανομή πόρων, ο σχεδιασμός μιας διάταξης και η υπολογιστική όραση. Στην πληροφορική, το CSP χρησιμοποιείται ως μια σημαντική προσέγγιση για την επίλυση προβλημάτων. Η πληροφορική περιλαμβάνει την ανάλυση δεδομένων, την ανάπτυξη αλγορίθμων και τον σχεδιασμό λογισμικού και συστημάτων, όπου το CSP μπορεί να αξιοποιηθεί για την ανάπτυξη αποδοτικών λύσεων. Για παράδειγμα, το CSP μπορεί να χρησιμοποιηθεί σε συστήματα βάσεων δεδομένων όταν χρειάζεται να οριστούν περιορισμοί για τη διατήρηση της συνέπειας των δεδομένων και να βρεθούν λύσεις που να τους ικανοποιούν.

### Υπολογιστική Σκέψη

Αυτό το πρόβλημα είναι ένα πρόβλημα ικανοποίησης περιορισμών, που σημαίνει ότι για να το λύσετε, πρέπει να οργανώσετε τις συνθήκες που θέλει κάθε κάστορας, εφαρμόζοντας λογική σκέψη. Στη συνέχεια, πρέπει να λύσουμε τις επιλογές που περιορίζονται από τους περιορισμούς. Υποθέτοντας ότι μπορείτε να ικανοποιήσετε όλους τους κάστορες, ξεκινάτε ικανοποιώντας πρώτα εκείνον που έχει τις πιο αυστηρές συνθήκες. Έπειτα, ικανοποιείτε την επόμενη πιο αυστηρή συνθήκη, και συνεχίζετε έτσι μέχρι να ικανοποιηθούν όλες οι συνθήκες. Αυτή η διαδικασία επιτρέπει στους μαθητές να βιώσουν τη διαδικασία διάσπασης ενός σύνθετου προβλήματος σε μικρότερα υπο-προβλήματα, να ορίσουν περιορισμούς για κάθε υπο-πρόβλημα και στη συνέχεια να βρουν μια λύση που να ικανοποιεί αυτούς τους περιορισμούς.

## Χώρα προέλευσης θέματος

Νότια Κορέα

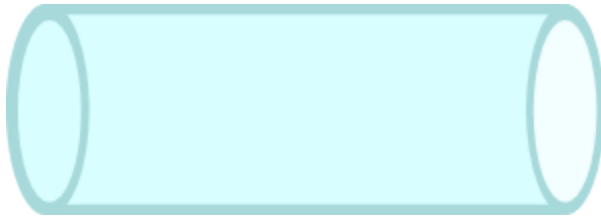
## 2024-KR-03a-GR - Σωλήνας

Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

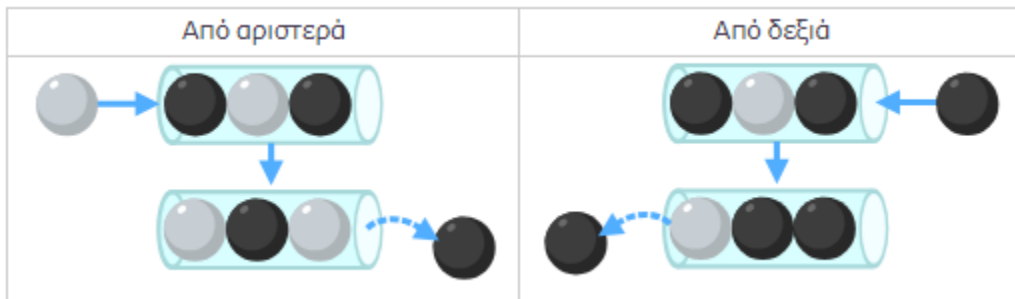
Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Σωλήνας

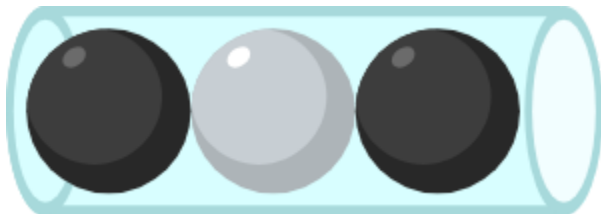
Έχουμε έναν σωλήνα με τρύπες και στις δύο πλευρές του, που μπορεί να γεμίσει με μέχρι 3 μπάλες.



Οι επόμενες εικόνες δείχνουν τι συμβαίνει όταν τρεις μπάλες βρίσκονται μέσα στον σωλήνα και προσπαθούμε να βάλουμε μια τέταρτη μπάλα μέσα σ' αυτόν.



Ο σωλήνας τώρα έχει γεμίσει έτσι:



Στη συνέχεια, τέσσερις ακόμη μπάλες σπρώχνονται μέσα στο σωλήνα η μία μετά την άλλη:



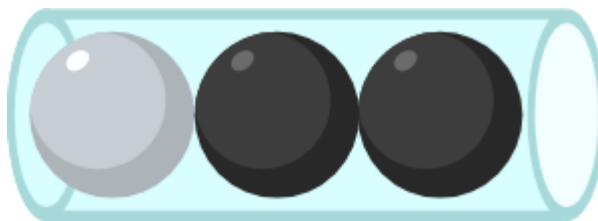
### Ερώτηση

Ποιες τρεις μπάλες βρίσκονται μέσα στον σωλήνα στο τέλος; Σύρετε τις μπάλες στη σωστή θέση μέσα στο σωλήνα.

### Απάντηση

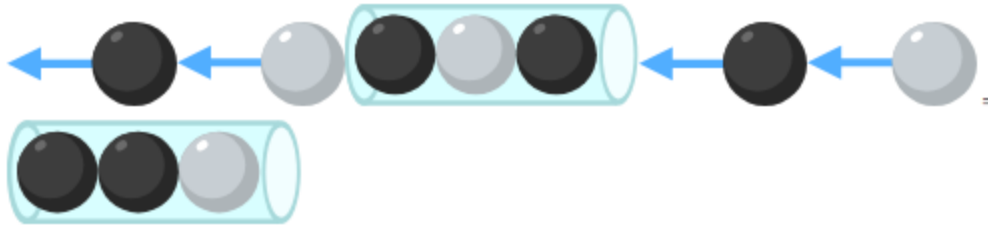


### Εξήγηση απάντησης

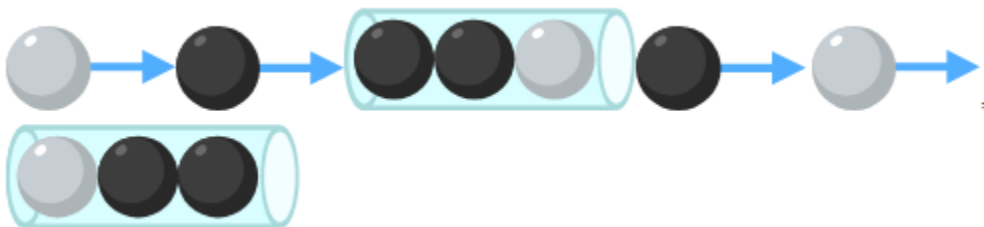


Η σωστή απάντηση είναι:

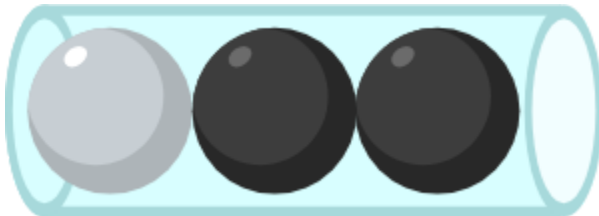
Δύο μπάλες σπρώχνονται στον σωλήνα από τα δεξιά, πρώτα μια μαύρη μπάλα και μετά μια λευκή μπάλα. Επομένως, δύο μπάλες πέφτουν στα αριστερά, πρώτα η μαύρη και μετά η άσπρη.



Στη συνέχεια, δύο μπάλες σπρώχνονται στον σωλήνα από τα αριστερά, πρώτα μια μαύρη μπάλα και μετά μια λευκή μπάλα. Αυτή τη φορά λοιπόν δύο μπάλες πέφτουν στα δεξιά, πρώτα η λευκή και μετά η μαύρη.



Επομένως η σωστή απάντηση είναι:



Η λύση μπορεί να βρεθεί και πιο απλά: αφού δύο μπάλες σπρώχνονται πρώτα στον σωλήνα από τα δεξιά και μετά δύο μπάλες από τα αριστερά, η μπάλα που ήταν αρχικά δεξιά στο σωλήνα είναι τώρα στα δεξιά και στα αριστερά του αυτό (από δεξιά προς τα αριστερά) είναι οι δύο μπάλες που ωθήθηκαν τελευταία στον σωλήνα.

Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Ο σωλήνας αυτής της άσκησης είναι μια ειδική μορφή “ουράς”(queue), η οποία στην επιστήμη των υπολογιστών ονομάζεται deque (συντομογραφία για ουρά διπλού άκρου). Σε αντίθεση με μια παραδοσιακή “ουρά”, μπορούμε να προσθέσουμε και να αφαιρέσουμε ένα στοιχείο και από τις δύο άκρες της. Το deque είναι μια δομή δεδομένων. Τα Deques έχουν πολλές χρήσεις, για παράδειγμα ως γενικευμένη αντικατάσταση τόσο για στοίβες (προσθήκη μπροστά και αφαίρεση μπροστά) όσο και για απλές ουρές (προσθήκη στο πίσω μέρος και αφαίρεση στο μπροστινό μέρος). Μπορείτε να το σκεφτείτε σαν μια αμαξοστοιχία: βαγόνια μπορούν να προστεθούν ή να αφαιρεθούν μπροστά και πίσω, αλλά δεν μπορείτε απλά να αφαιρέσετε ένα βαγόνι από τη μέση. Σε μια απλή αμαξοστοιχία, από την άλλη πλευρά, θα μπορούσατε να προσθέσετε ή να αφαιρέσετε βαγόνια μόνο στη μία πλευρά (σαν στοίβα) και σε έναν σταθμό, τα βαγόνια κινούνται μόνο προς μία κατεύθυνση (σαν ουρά). Ακριβώς όπως ο σωλήνας αυτής της άσκησης μπορεί να χωρέσει μόνο έως τρεις μπάλες, ένα deque έχει επίσης περιορισμένη χωρητικότητα στην πράξη. Εάν ένα στοιχείο προστεθεί σε ένα deque που είναι ήδη σε χωρητικότητα, πρέπει να δημιουργηθεί χώρος στο άλλο άκρο. Αυτό γίνεται αφαιρώντας ένα στοιχείο από εκεί και χάνοντας τις αντίστοιχες πληροφορίες. Ο χώρος αποθήκευσης είναι ένας πεπερασμένος πόρος και ο όγκος των δεδομένων που μπορείτε να αποθηκεύσετε εξαρτάται από αυτόν. Ωστόσο, τα deques χρησιμοποιούνται συχνά ως υποθετικά απεριόριστα, ειδικά όταν ασχολούνται με θεωρητικά ερωτήματα παρά με πρακτικές εφαρμογές.

## Χώρα προέλευσης θέματος

Νότια Κορέα

## 2024-MT-02-GR - Δωρεάν θέσεις στάθμευσης

Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Δωρεάν θέσεις στάθμευσης

Ο Φίλιππος είναι ένας κάστορας που βοηθάει να σταθμεύουν οι βάρκες στο λιμάνι. Πρέπει να βρίσκει κενές θέσεις στάθμευσης για βάρκες που θέλουν να μείνουν για δύο συνεχόμενες ημέρες, είτε Παρασκευή, Σάββατο ή Κυριακή. Παρακάτω είναι μια εικόνα που δείχνει πού σταθμεύουν οι βάρκες αυτές τις ημέρες



### Ερώτηση

Επιλέξτε τις θέσεις στάθμευσης που θα είναι κενές για τουλάχιστον δύο συνεχόμενες ημέρες.

### Απάντηση

1	2
3	4
5	6
7	8
9	10

## Εξήγηση απάντησης

Η σωστή απάντηση είναι οι ακόλουθες θέσεις στάθμευσης:

Ο χώρος στάθμευσης 1 είναι ελεύθερος και για τις τρεις ημέρες. Ο χώρος στάθμευσης 2 είναι ελεύθερος από Σάββατο έως Κυριακή. Ο χώρος στάθμευσης 5 είναι ελεύθερος και για τις τρεις ημέρες. Ο χώρος στάθμευσης 7 είναι ελεύθερος από Παρασκευή έως Σάββατο. Όλες οι άλλες θέσεις στάθμευσης δεν είναι ελεύθερες για δύο συνεχόμενες ημέρες ή δεν είναι καθόλου ελεύθερες κατά τη διάρκεια των τριών ημερών.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Αυτή η εργασία περιλαμβάνει το σχεδιασμό ενός λογικού κυκλώματος για την επεξεργασία δυαδικών δεδομένων που αντιπροσωπεύουν την κατάσταση των κόλπων της μαρίνας γιοτ. Απαιτεί την κατανόηση των λογικών πυλών (AND, OR, NOT) και πώς μπορούν να συνδυαστούν για την ανάλυση δυαδικών ακολουθιών. Στην επιστήμη των υπολογιστών, παρόμοιες λογικές πύλες και κυκλώματα χρησιμοποιούνται στην αρχιτεκτονική των υπολογιστών, στην επεξεργασία ψηφιακών σημάτων και σε διάφορα άλλα πεδία για την επεξεργασία και την ανάλυση δυαδικών δεδομένων. Με την επίλυση αυτής της εργασίας, οι μαθητές μπορούν να βελτιώσουν τη λογική τους σκέψη και τις δεξιότητες επίλυσης προβλημάτων, οι οποίες είναι θεμελιώδεις στον τομέα της επιστήμης των υπολογιστών.

Υπολογιστική Σκέψη

Αυτή η άσκηση περιλαμβάνει βασικές έννοιες Υπολογιστικής Σκέψης. Πρώτον, για να λύσει αυτή την εργασία ένας μαθητής εμπλέκεται στην Αναγνώριση Μοτίβου περιλαμβάνει τη διάκριση ποιες θέσεις είναι διαθέσιμες κάθε μέρα και την ανίχνευση της αλληλουχίας των κατειλημμένων και των μη κατειλημμένων θέσεων κατά τη διάρκεια των τριών ημερών. Στη συνέχεια, ο μαθητής ασχολείται με το Decomposition τμηματοποιώντας το ευρύτερο ζήτημα σε μικρότερα, πιο διαχειρίσιμα στοιχεία, τα οποία



περιλαμβάνουν την εξέταση κάθε θέσης κατά τη διάρκεια των τριών ημερών ξεχωριστά, σε αντίθεση με την προσπάθεια επίλυσης της κατάστασης των θέσεων ολόκληρης της μαρίνας με μία κίνηση. Στη συνέχεια, ο εκπαιδευόμενος εφαρμόζει Λογικό Συλλογισμό διατυπώνοντας μια συστηματική προσέγγιση για την αντιμετώπιση του προβλήματος και τέλος, αξιολογεί επιλέγοντας από τις επιλογές που του παρουσιάζονται.

## Χώρα προέλευσης θέματος

Μάλτα

## 2024-DE-03 – Το λουλούδι θαύμα



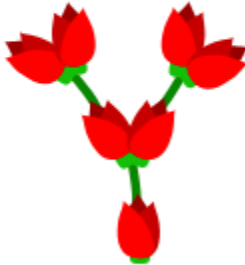
Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

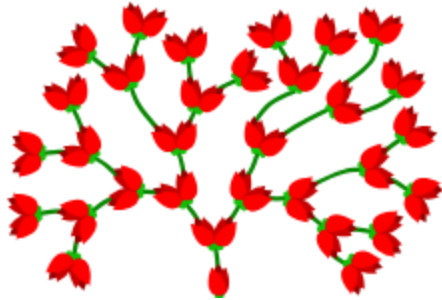
Θέμα : Το λουλούδι θαύμα

Στην αρχή της ημέρας, αρχίζει να αναπτύσσεται ένας μίσχος (δηλαδή το κοτσάνι του λουλουδιού) με δύο νέα μπουμπούκια από κάθε νέο μπουμπούκι του θαυματουργού λουλουδιού.

Στο τέλος της ημέρας, δύο νέα μπουμπούκια αναπτύσσονται στο τέλος κάθε μίσχου. Αυτό συνεχίζεται μέρα με τη μέρα και το θαυματουργό λουλούδι γίνεται όλο και πιο υπέροχο.

Αρχή της 1ης μέρας	Τέλος της 1ης μέρας	Τέλος της 2ης μέρας
		

Μετά από πολλές μέρες το θαυματουργό λουλούδι έχει μεγαλώσει όπως φαίνεται παρακάτω:



## Ερώτηση

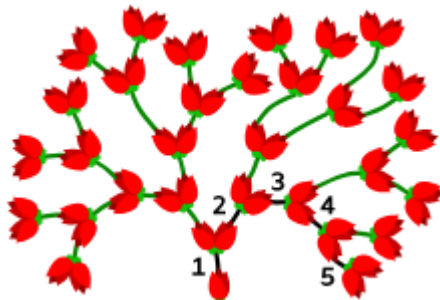
Για πόσες μέρες μεγάλωνε το λουλούδι για να γίνει έτσι;

## Απάντηση

5

## Εξήγηση απάντησης

Η σωστή απάντηση είναι 5 ημέρες. Στο τέλος κάθε μέρας, κάθε μπουμπούκι στο άκρο ενός μίσχου του θαυματουργού λουλουδιού θα έχει μεγαλώσει ένα νέο μίσχο με δύο νέα μπουμπούκια. Απλώς πρέπει να ακολουθήσετε ένα κλαδί, όπως αυτό στην άκρη δεξιά, για να μετρήσετε το ο αριθμός των ημερών αυξήθηκε. Αυτό φαίνεται στο παρακάτω



διάγραμμα:

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Τα θαυματουργά λουλούδια αναπτύσσονται σύμφωνα με έναν συγκεκριμένο κανόνα: κάθε μέρα, κάθε νέο μπουμπούκι σχηματίζει ακριβώς ένα στέλεχος και δύο νέα μπουμπούκια. Την επόμενη μέρα, ο ίδιος κανόνας εφαρμόζεται σε όλα τα νεοσχηματισμένα μπουμπούκια. Αυτή είναι μια συνεχής διαδικασία που εφαρμόζει επανειλημμένα τον ίδιο κανόνα. Θεωρητικά, το θαυματουργό λουλούδι μπορεί να συνεχίσει να αναπτύσσεται επ'αόριστον. Αυτή η επαναλαμβανόμενη εφαρμογή ενός κανόνα είναι κοινή σε συστήματα ελεγχόμενα από υπολογιστή. Στην επιστήμη των υπολογιστών, η διαδικασία επανάληψης των ίδιων ή παρόμοιων ενεργειών πολλές φορές ονομάζεται επανάληψη. Για παράδειγμα, τα χειριστήρια των φωτεινών σηματοδοτών δημιουργούν συνεχώς την ίδια σειρά φαναριών. Σε έναν κουλοχέρη, το ίδιο παιχνίδι μπορεί να επανεκκινηθεί ξανά και ξανά.

### Υπολογιστική σκέψη

Η έννοια της αφαίρεσης μας οδηγεί να εστιάζουμε μόνο στις πιο σημαντικές πληροφορίες και στοιχεία του προβλήματος, αγνοώντας οτιδήποτε άλλο, ειδικά άσχετες ή περιττές λεπτομέρειες. Αυτή η εργασία απαιτεί από τον μαθητή να καταλάβει ότι αρκεί να κοιτάξει απλώς το μήκος οποιουδήποτε κλάδου.

### Χώρα προέλευσης θέματος

Γερμανία

## 2024-PL-04 – Η διαδρομή του ρομπότ

Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Η διαδρομή του ρομπότ

Το ρομπότ του Βασίλη κινείται σε πλακίδια ακολουθώντας ένα σύνολο κανόνων:

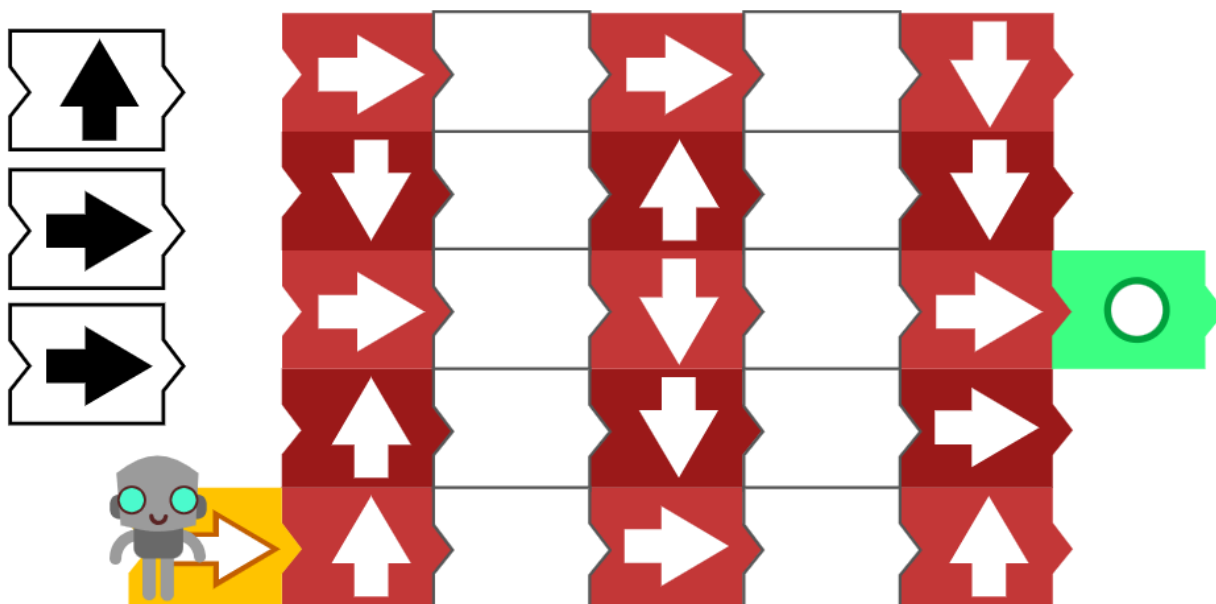
- Το ρομπότ ξεκινά από το πιο αριστερό πλακίδιο, όπως βλέπετε στην παρακάτω εικόνα.
- Το ρομπότ κινείται προς την κατεύθυνση του βέλους στο οποίο στέκεται.
- Εάν το ρομπότ απομακρυνθεί από τα πλακίδια, σταματά.

Ορισμένα πλακίδια είναι προκαθορισμένα όπως φαίνεται παρακάτω. Στον Βασίλη δίνονται τρία επιπλέον πλακίδια.

### Ερώτηση

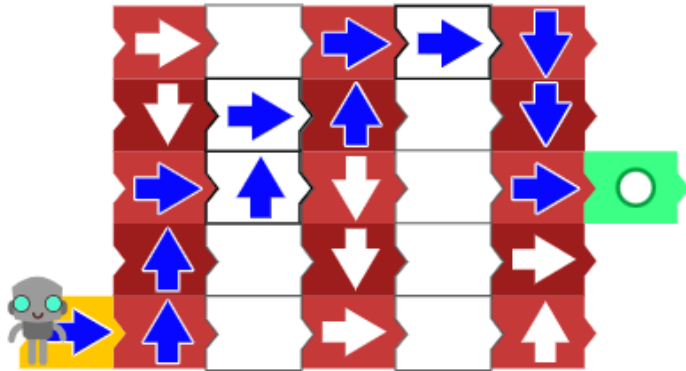
Βοηθήστε τον Βασίλη να τοποθετήσει και τα τρία επιπλέον πλακίδια στις κενές περιοχές, έτσι ώστε το ρομπότ του να φτάσει στο πράσινο πλακίδιο με τον κύκλο.

*Σύρετε τα βέλη στη σωστή θέση. Κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.*



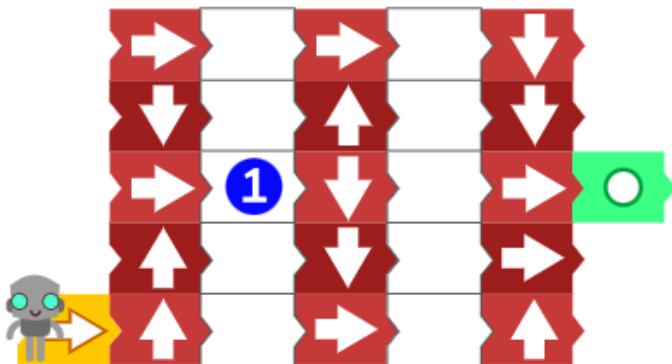
## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Όπως έχει ήδη υποδειχθεί, το ρομπότ του Βασίλη θα κινηθεί πρώτα προς τα δεξιά, δύο φορές προς τα πάνω και πάλι προς τα δεξιά. Αυτό σημαίνει ότι μια κάρτα πρέπει να τοποθετηθεί στη θέση που σημειώνεται με το 1:



Οι δύο τύποι διαθέσιμων καρτών είναι «πάνω» ή «δεξιά». Αν επιλέξετε «δεξιά», το ρομπότ θα κινηθεί προς τα δεξιά, μετά δύο φορές προς τα κάτω και πάλι προς τα δεξιά. Τώρα θα μπορούσατε είτε να επιλέξετε πάλι «δεξιά» είτε «πάνω» και μετά «δεξιά». Και στις δύο περιπτώσεις, το ρομπότ θα προσγειωνόταν κάτω από τον επιθυμητό στόχο:



επιστρέψετε και θα δοκιμάσετε έναν άλλο τρόπο.

Στην επιστήμη των υπολογιστών, αυτό ονομάζεται αναζήτηση σε βάθος (depth-first search), επειδή πρώτα δοκιμάζεται εντελώς ένας δρόμος και μετά δοκιμάζεται ο επόμενος. Επειδή συχνά πρέπει να πάτε ένα ή περισσότερα βήματα πίσω για να δοκιμάσετε μια διαφορετική λύση, κάνετε backtracking. Σε κάθε περίπτωση, όμως, λύνετε το πρόβλημα με την προς τα εμπρός ανίχνευση.

Ωστόσο, ορισμένα προβλήματα επιλύονται καλύτερα προς τα πίσω, κοιτάζοντας από τον στόχο για να δείτε πώς θα μπορούσατε να φτάσετε εκεί. Στην πραγματικότητα, αυτό το είδος προβλήματος δεν είναι κατάλληλο για να λυθεί προς τα πίσω, διότι ενώ όταν πηγαίνετε προς τα εμπρός ξέρετε πάντα ποια κάρτα είναι η επόμενη, όταν πηγαίνετε προς τα πίσω μπορεί να σημαίνει ότι έρχεστε από πολλές κάρτες (το βέλος δείχνει προς τα εμπρός και πηγαίνοντας προς τα πίσω πρέπει να ψάξετε όλες τις κάρτες γύρω του για να δείτε αν δείχνουν προς την κατεύθυνση της κάρτας).

Αλλά στη συγκεκριμένη περίπτωση, εξακολουθεί να είναι εύκολο να δείτε ποια λύση μπορεί να είναι η μόνη σωστή.

Υπολογιστική σκέψη

Ένας υπολογιστής είναι καλός στο να δοκιμάζει απλώς όλες τις πιθανές περιπτώσεις. Δεδομένου ότι υπάρχουν μόνο (για έναν υπολογιστή) πολύ λίγες πιθανές θέσεις και πολύ λίγες πιθανές κάρτες διαθέσιμες, ένας υπολογιστής μπορεί πολύ γρήγορα να δοκιμάσει απλά όλες τις πιθανότητες. Ένας τέτοιος σχετικά «αφελής» αλγόριθμος θα έλυne καλά το πρόβλημα.

Ωστόσο, οι άνθρωποι χρειάζονται λίγο περισσότερο χρόνο για να δοκιμάσουν τα πάντα. Επομένως, αξίζει να αναλύσουμε το πρόβλημα πιο προσεκτικά και να διατυπώσουμε περιορισμούς που μειώνουν κατά πολύ τον αριθμό των περιπτώσεων που πρέπει να δοκιμαστούν. Για παράδειγμα, μια κάρτα πρέπει να βρίσκεται στην τρίτη υψηλότερη θέση της αριστερής κόκκινης στήλης. Αυτό περιορίζει σημαντικά τον χώρο αναζήτησης πιθανών λύσεων, έτσι ώστε τελικά να χρειάζεται να δοκιμαστούν μόνο δύο παραλλαγές, όπως φαίνεται στην επεξήγηση της απάντησης.

Η υπολογιστική σκέψη δεν σημαίνει ότι απλώς δοκιμάζουμε όλες τις πιθανές περιπτώσεις όπως κάνει ένας υπολογιστής, αλλά ότι λαμβάνουμε υπόψη απλουστεύσεις που χρησιμοποιούν και οι επιστήμονες της πληροφορικής. Σε αυτή την περίπτωση, αυτό θα μπορούσε να είναι μια εφαρμοσμένη αναζήτηση σε βάθος με οπισθοδρόμηση ή μια αναζήτηση προς τα πίσω αντί για αναζήτηση προς τα εμπρός.

## Χώρα προέλευσης θέματος

Πολωνία





## 2024-BG-01b – Μπάλες

Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Μπάλες

Παρακάτω βλέπετε μια ακολουθία από κόκκινες  και μπλε  μπάλες:



Για κάθε μπάλα, μετράμε τώρα πόσες μπλε μπάλες βρίσκονται στα δεξιά της (συμπεριλαμβανομένης της ίδιας της μπάλας) και το σημειώνουμε ως εξής:

3, 3, 2, 1, 1, 1.

Στο επόμενο βήμα, σημειώνουμε ξανά την ίδια ακολουθία. Αλλά αν ο αριθμός είναι ζυγός (άρτιος), σημειώνουμε ένα 0, αν είναι μονός (περιττός), τότε ένα 1. Αυτό οδηγεί σε αυτή την ακολουθία:

110111

### Ερώτηση

Δίνεται η παρακάτω ακολουθία των αριθμών 0 και 1:

01110100

Κατασκευάστε μια ακολουθία μπλε και κόκκινων μπάλων που να ταιριάζει με αυτή την αναπαράσταση.



### Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Υπάρχουν διάφοροι τρόποι για να βρείτε αυτή την απάντηση. Ένας τρόπος είναι να περάσετε την ακολουθία των αριθμών από τα δεξιά προς τα αριστερά και να συμπεράνετε λογικά αν η τρέχουσα μπάλα πρέπει να είναι κόκκινη ή μπλε.

Ας δούμε τις μπάλες με τη σειρά:

1. Ας κοιτάξουμε την πρώτη μπάλα στα δεξιά. Παίρνουμε το 0 σε αυτό το σημείο ως ένδειξη ότι η μπάλα πρέπει να είναι κόκκινη, επειδή μόνο μια κόκκινη μπάλα διατηρεί το άθροισμα όλων των μπλε μπαλών ίσο.

0	1	1	1	0	1	0	0
							Κόκκινη

2. Στη δεύτερη θέση από τα δεξιά βρίσκουμε ένα 0. Το επιχείρημα παραμένει το ίδιο όπως και πριν: μόνο με μια κόκκινη μπάλα το άθροισμα όλων των μπλε μπαλών παραμένει σωστό και συμπεριλαμβανομένης αυτής της θέσης αυτή τη στιγμή.

0	1	1	1	0	1	0	0
						Κόκκινη	Κόκκινη

3. Στην τρίτη θέση από τα δεξιά βρίσκουμε ένα 1, το οποίο υποδηλώνει ότι ο αριθμός των μπλε μπαλών θα πρέπει να είναι μονός από αυτό το σημείο και μετά. Εφόσον γνωρίζουμε ήδη ότι οι δύο μπάλες στα δεξιά είναι κόκκινες, άρα η τρέχουσα μπάλα πρέπει να είναι μπλε για να μετακινηθούμε από έναν ευθύγραμμο σε έναν μονό αριθμό μπλε μπαλών.

0	1	1	1	0	1	0	0
					Μπλε	Κόκκινη	Κόκκινη

4. Στην τέταρτη θέση από δεξιά βρίσκουμε ένα 0. Για να αντικαταστήσουμε τις προηγούμενες μονές μπάλες (2x κόκκινες και 1x μπλε) για να πάρουμε έναν ζυγό αριθμό μπλε μπαλών, ο αριθμός των μπλε μπαλών της τέταρτης μπάλας πρέπει να είναι μπλε.

0	1	1	1	0	1	0	0
				Μπλε	Μπλε	Κόκκινη	Κόκκινη

5. Στην πέμπτη θέση από δεξιά βρίσκουμε ένα 1. Αφού το προηγούμενο άθροισμα των μπλε μπαλών ήταν ζυγό και τώρα θα είναι μονό, η νέα μπάλα πρέπει να είναι επίσης μπλε.

0	1	1	1	0	1	0	0
			Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

6. Στην έκτη θέση από δεξιά βρίσκεται ο αριθμός 1. Εφόσον ο προηγούμενος αριθμός μπλε μπαλών ήταν περιττός, η τρέχουσα μπάλα πρέπει να είναι κόκκινη, ώστε ο αριθμός που εμφανίζεται να είναι περιττός.

0	1	1	1	0	1	0	0
		Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

7. Στην έβδομη θέση από τα δεξιά βρίσκουμε άλλο ένα 1. Η τρέχουσα ακολουθία περιέχει ήδη έναν περιττό αριθμό μπλε μπαλών- για να διατηρηθεί αυτή η κατάσταση, η νέα μπάλα πρέπει να έχει κόκκινο χρώμα.

0	1	1	1	0	1	0	0
	Κόκκινη	Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

8. Στην όγδοη θέση από τα δεξιά (πρώτη θέση από τα αριστερά) βρίσκουμε τον αριθμό 0. Για να γίνει και πάλι ζυγός ο μονός αριθμός μπλε μπαλών, πρέπει να είναι μια μπλε μπάλα.

0	1	1	1	0	1	0	0
Μπλε	Κόκκινη	Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

Συνοπτικά, σε κάθε μετάβαση από το 0 στο 1 ή από το 1 στο 0 πρέπει να χρησιμοποιείται μια μπλε μπάλα, ενώ μια συνεχής ακολουθία των ίδιων αριθμών υποδεικνύει κόκκινες μπάλες. Αυτές δεν επηρεάζουν το άθροισμα των μπλε μπαλών:

0	1	1	1	0	1	0	0
4 Μπλε	3	3	3 Μπλε	2 Μπλε	1 Μπλε	0	0
	Κόκκινες	Κόκκινες				Κόκκινες	Κόκκινες

Τότε η ακολουθία των μπλε μπαλών είναι 43332100.



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

- Η κωδικοποίηση χρησιμοποιείται για την αναπαράσταση πολύπλοκων θεμάτων με τρόπο ώστε οι υπολογιστές να μπορούν εύκολα να τα επεξεργαστούν. Συχνά άσχετα δεδομένα μπορούν να παραλειφθούν χωρίς να επηρεαστούν τα κεντρικά στοιχεία της πληροφορίας. Αυτή η εργασία προτείνει μια κωδικοποίηση του τρόπου με τον οποίο μπορεί να αποθηκευτεί ο ακριβής αριθμός και η θέση των μπλε και κόκκινων μπαλών χωρίς τα ίδια τα πραγματικά χρώματα. Λόγω της επιλεγμένης κωδικοποίησης βρισκόμαστε στη θέση της αρχικής πληροφορίας (τα χρώματα και η θέση των αρχικών μπαλών).
- Υπό ορισμένες συνθήκες, η κωδικοποίηση μπορεί επίσης να εξοικονομήσει αποθηκευτικό χώρο, κάτι που ονομάζουμε συμπίεση. Για την αναπαράσταση δυαδικής πληροφορίας (όπως στην πληροφορία αν ένας αριθμός είναι ζυγός ή περιττός) αρκεί ένα μόνο bit πληροφορίας, ενώ για την αποθήκευση οποιουδήποτε αριθμού ή χρώματος απαιτείται περισσότερος αποθηκευτικός χώρος.
- Ως πρώτο ενδιάμεσο βήμα, προσδιορίστηκε μια αθροιστική συχνότητα για τον υπολογισμό του αθροίσματος των στοιχείων που έχουν ένα συγκεκριμένο χαρακτηριστικό (εδώ συγκεκριμένα τον αριθμό των μπλε μπαλών από την τρέχουσα θέση). Η αθροιστική συχνότητα είναι ένα εργαλείο της στατιστικής, το οποίο χαρακτηρίζεται ιδιαίτερα από το γεγονός ότι η πληροφορία που προκύπτει συσσωρεύεται (εδώ όχι μόνο μια ένδειξη για το χρώμα της τρέχουσας μπάλας, αλλά και για τα χρώματα των μπαλών πίσω από αυτήν). Οι λύσεις προβλημάτων που χρησιμοποιούν τη σωρευτική συχνότητα συχνά επωφελούνται από το γεγονός ότι μπορείτε να επεξεργαστείτε πολλές αιτήσεις γρήγορα χωρίς να χρειάζεται να έχετε πρόσβαση στα αρχικά δεδομένα.

### Υπολογιστική σκέψη

Κατά τη διάρκεια της διαδικασίας κωδικοποίησης ενός προβλήματος, οι μαθητές εξασκούν την ικανότητα της αφαίρεσης: παραλείπονται οι άσχετες πληροφορίες και γίνεται σταδιακή μείωση των λεπτομερειών, χωρίς να χάνεται το κεντρικό πληροφοριακό περιεχόμενο.

## Χώρα προέλευσης θέματος

Βουλγαρία

## 2024-BE-01a – Τουβλάκια στην σειρά

Ηλικιακή ομάδα: Ε' – ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

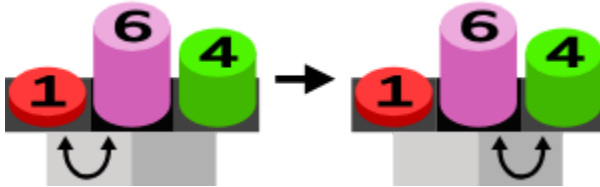
Θέμα : Τουβλάκια στην σειρά

Ένα μηχάνημα μπορεί να μετακινήσει έξι τουβλάκια διαφορετικού ύψους. Χρησιμοποιεί ένα βελάκι που τοποθετείται ανάμεσα σε δύο τουβλάκια. Στην αρχή, τα τουβλάκια και το βελάκι βρίσκονται σε μια αρχική θέση, για παράδειγμα έτσι:



Μετά, το μηχάνημα εκτελεί συνέχεια ένα από αυτά τα δύο βήματα:

Εάν το τουβλάκι στα αριστερά του βέλους είναι χαμηλότερο από το τουβλάκι στα δεξιά του βέλους, τότε το όλο βελάκι μετακινείται προς τα δεξιά.



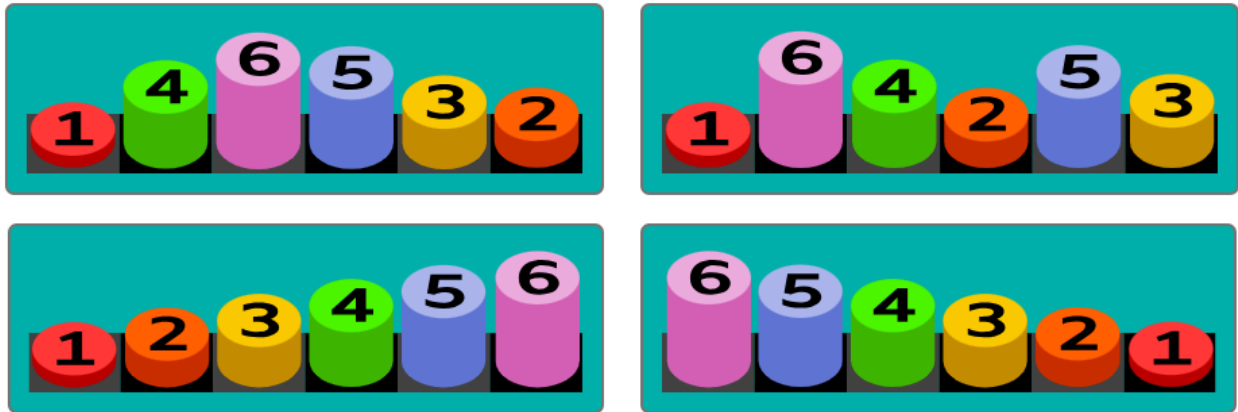
Εάν το τουβλάκι στα αριστερά του βέλους είναι ψηλότερο από το τουβλάκι στα δεξιά του βέλους, τα τουβλάκια αλλάζουν θέση μεταξύ τους. Επιπλέον, όλο το βελάκι μετακινείται προς τα αριστερά - μόνο αν υπάρχει διαθέσιμη γκρι θέση.



Το μηχάνημα σταματά όταν το βελάκι φτάσει στην τελευταία θέση δεξιά.

## Ερώτηση

Πού βρίσκονται τα τουβλάκια όταν σταματήσει το μηχάνημα;



## Απάντηση



## Εξήγηση απάντησης

Η σωστή απάντηση είναι:



Βασική ιδέα: Όλα τα τουβλάκια στα αριστερά του δείκτη είναι πάντα ταξινομημένα. Η αρχή είναι ξεκάθαρη και τα δύο βήματα το διατηρούν. Φυσικά, θα μπορούσατε να παρακολουθήσετε τη δουλειά του μηχανήματος βήμα προς βήμα, αλλά είναι πιο εύκολο να δείτε ότι το μηχάνημα ταξινομεί τα τουβλάκια ανάλογα με το μέγεθος. Η σήμανση μετακινείται προς τα δεξιά μόνο εάν το τουβλάκι στα αριστερά της σήμανσης είναι μικρότερο από το τουβλάκι στα δεξιά της. Εάν ένα μικρότερο τουβλάκι αντικατασταθεί προς τα αριστερά, η σήμανση μετακινείται επίσης προς τα αριστερά. Αυτό καθιστά σαφές ότι στα αριστερά της σήμανσης, όλα τα τουβλάκια είναι ήδη ταξινομημένα με αύξουσα σειρά μεγέθους. Ένα μικρότερο τουβλάκι ανταλλάσσεται πιο αριστερά μέχρι να γίνει μεγαλύτερο από το τουβλάκι στα αριστερά του (και επομένως από όλα τα άλλα αριστερά του). Στη συνέχεια, η σήμανση προς τα δεξιά μέχρι να εμφανιστεί ένα άλλο

μικρότερο τουβλάκι . Με αυτόν τον τρόπο, τα δομικά στοιχεία ταξινομούνται βήμα προς βήμα ανάλογα με το μέγεθος, σε αύξουσα σειρά. Μόλις όλα τα τουβλάκια ταξινομηθούν με αύξουσα σειρά μεγέθους, ο δείκτης μετακινείται βήμα προς βήμα προς τα δεξιά μέχρι να βρεθεί στα δεξιά των τούνλων και το μηχάνημα να σταματήσει. Στη συνέχεια, τα τουβλάκια ταξινομούνται κατά μέγεθος σε αύξουσα σειρά, όπως στην απάντηση.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Η ταξινόμηση των δεδομένων είναι μια κοινή διαδικασία στην επιστήμη των υπολογιστών, είτε κατά μέγεθος, είτε κατά κόστος είτε αλφαβητικά. Επομένως, ένα από τα καθήκοντα των επιστημόνων υπολογιστών είναι να βρουν τις πιο αποτελεσματικές μεθόδους ταξινόμησης. Η μέθοδος ταξινόμησης σε αυτήν την εργασία Bebras ονομάζεται ταξινόμηση Gnome. Είναι παρόμοια με τη μέθοδο της φυσαλίδας. Δεν είναι ιδιαίτερα αποτελεσματική. Οι επιστήμονες υπολογιστών αξιολογούν τη μέση απόδοση χρόνου εκτέλεσης ως σημαντικά χειρότερη σε σύγκριση με αποτελεσματικές μεθόδους ταξινόμησης όπως η γρήγορη ταξινόμηση ή η συγχώνευση. Ωστόσο, η ταξινόμηση Gnome είναι πιο κατανοητή.

## Χώρα προέλευσης θέματος

Βέλγιο



## 2024-SI-01 – Επιστροφή βιβλίων

Ηλικιακή ομάδα: Ε' - ΣΤ' Δημοτικού

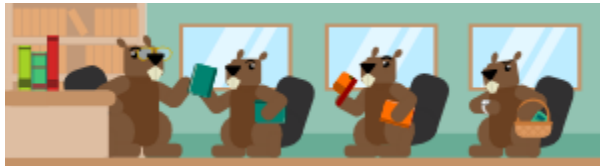
Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Επιστροφή βιβλίων

Επόμενος παρακαλώ! Οι κάστορες στο Holzdorf είναι μανιώδεις αναγνώστες. Για αυτόν τον λόγο υπάρχουν συνήθως πολλοί κάστορες στην ουρά στο γραφείο επιστροφής βιβλίων της βιβλιοθήκης. Ισχύει ο ακόλουθος κανόνας:

Ο κάστορας με τα λιγότερα βιβλία είναι ο επόμενος στην ουρά.

Η βιβλιοθηκάριος χρειάζεται πάντα ακριβώς ένα λεπτό για να επιστρέψει ένα βιβλίο. Μόλις σημειώσει τα βιβλία που επέστρεψε ο ένας κάστορας και τα βάλει στην άκρη, ο επόμενος κάστορας από την ουρά παίρνει τη σειρά του. Ο επόμενος κάστορας είναι πάντα αυτός με τα λιγότερα βιβλία.



Ένα πρωί έρχονται 5 κάστορες και θέλουν να επιστρέψουν τα βιβλία τους. Ο πίνακας δείχνει την ώρα άφιξης και τον αριθμό των βιβλίων για κάθε κάστορα.

Όνομα	Ωρα άφιξης	Αριθμός βιβλίων
Ana	9:00	4
Beti	9:02	6
Cene	9:03	2
Darja	9:05	4
Emil	9:11	1

Η Ana φτάνει πρώτη και μπορεί να επιστρέψει αμέσως τα 4 βιβλία της.

### Ερώτηση

Με ποια σειρά επιστρέφουν οι κάστορες τα βιβλία τους;

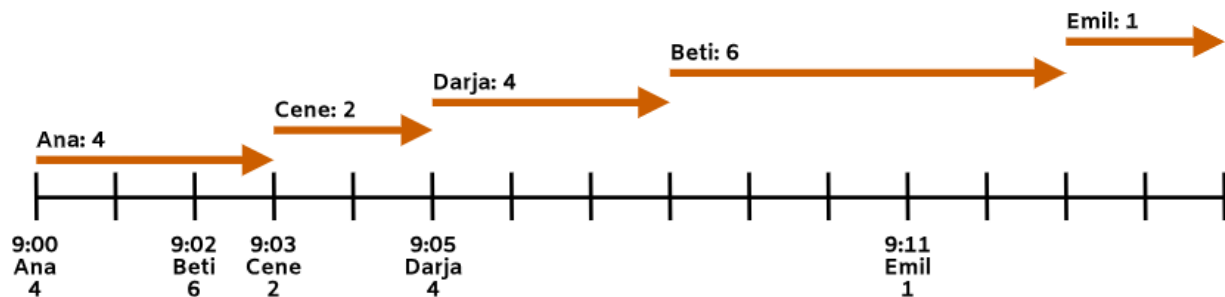
## Απάντηση

1. Ana
2. Cene
3. Darja
4. Beti
5. Emil

## Εξήγηση απάντησης

Η σωστή απάντηση είναι Ana, Cene, Darja, Beti, Emil.

Το παρακάτω διάγραμμα Gantt παρέχει μια οπτική αναπαράσταση του πώς ο βιβλιοθηκάριος αντιμετωπίζει αυτά τα βιβλία που επιστρέφουν και με ποια σειρά.



Αυτή η εικόνα δείχνει πώς επιστρέφονται τα βιβλία:

- Πρώτη η Ana. Χρειάζονται 4 λεπτά για να σημειώσει ο βιβλιοθηκάριος τα 4 βιβλία. Μόνο τότε μπορεί να λάβει τα επόμενα βιβλία.
- Στο μεταξύ, η Beti και η Cene έχουν φτάσει και περιμένουν. Η Cene έχει μόνο 2 βιβλία. Γι' αυτό είναι η επόμενη. Μετά από 2 λεπτά σημειώνονται τα βιβλία της.
- Στο μεταξύ, η Darja έχει έρθει και στέκεται δίπλα στον Beti. Η Darja έχει 4 βιβλία και η Beti έχει 6. Η Darja έχει τα λιγότερα βιβλία και είναι η επόμενη. Η καμμένη η Beti πρέπει ακόμα να περιμένει. Μετά από 4 λεπτά ο βιβλιοθηκάριος έχει τελειώσει με τα βιβλία της Darja και τώρα μπορεί επιτέλους να φροντίσει την Beti. Γιατί είναι η μόνη στην ουρά.
- Στο μεταξύ έφτασε και ο Emil. Είναι ο τελευταίος που πάει.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Η εργασία δείχνει πώς λειτουργεί ένας scheduler. Ο scheduler είναι ένα πρόγραμμα που ανήκει στο λειτουργικό σύστημα ενός υπολογιστή. Ενώ ένας υπολογιστής είναι σε λειτουργία, πολλές εργασίες πρέπει να υποβληθούν σε επεξεργασία.

Για κάθε εργασία, ξεκινά μια διαδικασία που λύνει την εργασία. Ο Scheduler ελέγχει τη ροή των διεργασιών. Καθορίζει πότε και για πόσο χρόνο εκτελείται μια διεργασία από την κεντρική μονάδα επεξεργασίας του υπολογιστή (CPU)

Υπάρχουν διαφορετικές στρατηγικές που χρησιμοποιούν οι scheduler:

- Priority Scheduling – Σε αυτόν τον τύπο ουράς, κάθε διεργασία έχει μια προτεραιότητα, και η CPU εκτελεί πρώτα τις διεργασίες με την υψηλότερη προτεραιότητα. Αν δύο διεργασίες έχουν την ίδια προτεραιότητα, χρησιμοποιείται μια δευτερεύουσα στρατηγική για την επιλογή.
- First Come First Serve (FCFS) – Οι διεργασίες εκτελούνται με τη σειρά που φτάνουν στην ουρά, χωρίς προτεραιότητες ή διακοπές. Αυτό σημαίνει ότι η πρώτη διεργασία που φτάνει στη CPU εκτελείται μέχρι να ολοκληρωθεί, προτού ξεκινήσει η επόμενη.
- Round Robin – Η CPU εκτελεί κάθε διεργασία για ένα συγκεκριμένο χρονικό διάστημα (χρονικό τεμάχιο - time quantum) και στη συνέχεια την τοποθετεί στο τέλος της ουράς αν δεν έχει ολοκληρωθεί, εξασφαλίζοντας δίκαιη κατανομή του χρόνου επεξεργασίας.
- Shortest Job First (SJF) – Η CPU εκτελεί πρώτα τη διεργασία με τον μικρότερο χρόνο εκτέλεσης, μειώνοντας τον μέσο χρόνο αναμονής. Αν δύο διεργασίες έχουν τον ίδιο χρόνο εκτέλεσης, επιλέγεται η πρώτη που έφτασε στην ουρά.

Υπολογιστική σκέψη

Σε αυτήν την εργασία, μας δίνεται ένα σύνολο κανόνων για το πώς λειτουργεί η βιβλιοθήκη, συμπεριλαμβανομένου του χρόνου που χρειάζεται ο βιβλιοθηκάριος για να επεξεργαστεί μια επιστροφή βιβλίου, πότε κάποιος μπορεί να φύγει από την ουρά και ποιος θα εξυπηρετείται στη συνέχεια. Αυτό το σύνολο κανόνων είναι στην πραγματικότητα ένας αλγόριθμος και ακολουθώντας προσεκτικά αυτόν τον αλγόριθμο στα δεδομένα που παρέχονται (η λίστα των δανειοληπτών βιβλίων και τα στοιχεία τους) μπορούμε να λύσουμε το πρόβλημα. Ο αλγόριθμος έχει πιθανές ασάφειες, όπως τι πρέπει να γίνει στην κατάσταση που δύο δανειολήπτες βιβλίων στην ουρά έχουν ταυτόχρονα τον ίδιο αριθμό βιβλίων, αλλά αυτή η κατάσταση δεν χρειάζεται να οριστεί για αυτό το συγκεκριμένο σενάριο.

## Χώρα προέλευσης θέματος

Σλοβενία

## 2024-TW-03-UK – Το πιο μακρύ βραχιόλι

Ηλικιακή ομάδα: Ε' - ΣΤ' Δημοτικού

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Το πιο μακρύ βραχιόλι

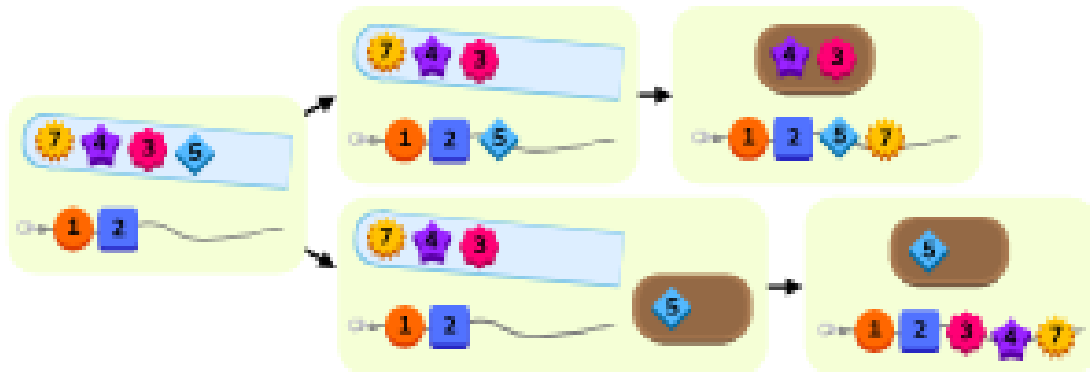
Ο Μάριος φτιάχνει ένα βραχιόλι. Παίρνει χάντρες με αριθμούς από ένα γυάλινο σωλήνα. Κολλάει μερικές χάντρες από αυτές και βάζει τις άλλες στο καφέ κουτί και δεν τις χρησιμοποιεί. Επιτρέπεται να περάσει μια χάντρα μόνο αν:

το κορδόνι είναι άδειο, ή

η χάντρα έχει μεγαλύτερο αριθμό από την τελευταία χάντρα στο κορδόνι.

Σε αυτό το παράδειγμα, η τελευταία χάντρα στο κορδόνι είναι η χάντρα 2. Στη συνέχεια, ο Μάριος μπορεί να περάσει τη χάντρα 5 από τον σωλήνα. Μπορεί επίσης να την αφήσει στο καφέ κουτί και να μην τη χρησιμοποιήσει.

Αν περάσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με τέσσερις χάντρες 1257. Αν δεν χρησιμοποιήσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με περισσότερες χάντρες 12347.



Ο Μάριος φτιάχνει ένα καινούργιο βραχιόλι από τις χάντρες αυτού του σωλήνα:



## Ερώτηση

Ποιος είναι ο μέγιστος αριθμός χαντρών που μπορεί να περάσει από το κορδόνι;

## Απάντηση

6

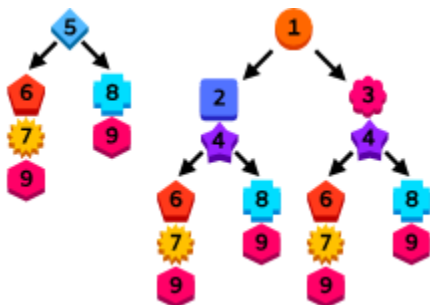
## Εξήγηση απάντησης

Η απάντηση είναι 6.

Θα μπορούσαμε να απαριθμήσουμε όλα τα βραχιόλια για αυτό το σωληνάριο και να δούμε ποια έχουν τον μεγαλύτερο αριθμό χαντρών. Αλλά αυτό θα ήταν χρονοβόρο. Ας ρίξουμε μια πιο προσεκτική ματιά στους αριθμούς:



Η πρώτη χάντρα είναι ο αριθμός 5. Όταν αυτή περνάει στο κορδόνι, οι επόμενες δυνατές χάντρες είναι μόνο οι 6, 7, 8 και 9, επομένως οι πιθανές ακολουθίες θα ήταν 5679 και 589. Εάν η χάντρα 5 αφηθεί στην άκρη και η επόμενη χάντρα, η χάντρα 1, μπει στο κορδόνι, υπάρχουν περισσότερες πιθανές ακολουθίες. Αυτό μπορεί να αναπαρασταθεί ξεκάθαρα σε ένα δέντρο αποφάσεων:



Εάν η χάντρα 1 αφηθεί επίσης στην άκρη και μία διαφορετική χάντρα περαστεί πρώτα, οι πιθανές αλληλουχίες περιέχονται ήδη σε μία από τις παραπάνω ακολουθίες. Εάν ξεκινήσετε με την χάντρα 2, για παράδειγμα, θα λάβετε 24679 ή 2489, που περιέχονται στις μεγαλύτερες ακολουθίες 124679 και 12489 αντίστοιχα.

Τα βραχιόλια με τον μεγαλύτερο αριθμό χαντρών ξεκινούν με τη χάντρα 1 και αποτελούνται από 6 χάντρες: 124679 ή 123679.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Κάθε σωλήνας περιέχει μια ακολουθία αριθμημένων χαντρών. Εάν ο Μάριος ακολουθήσει τον δεύτερο κανόνα (μόνο οι χάντρες στο κορδόνι που έχουν μεγαλύτερο αριθμό από την τελευταία), θα πάρει ένα βραχιόλι με αύξουσα ακολουθία. Για να κατασκευαστεί ένα βραχιόλι με όσο το δυνατόν περισσότερες χάντρες, πρέπει να καθορίσει η μεγαλύτερη ανοδική ακολουθία.

Για μεγάλες ακολουθίες, θα χρειαζόταν πολύς χρόνος για να σχηματιστούν όλες οι πιθανές αύξουσες υποακολουθίες και στη συνέχεια να προσδιοριστεί η μεγαλύτερη. Εάν ο σωλήνας περιείχε 20 χάντρες, για παράδειγμα, η υπολογιστική προσπάθεια θα ήταν της τάξης ενός εκατομμυρίου βημάτων.

Ευτυχώς, στην επιστήμη των υπολογιστών έχουν αναπτυχθεί αλγόριθμοι που μπορούν να βρουν τη μεγαλύτερη αύξουσα υποακολουθία πιο γρήγορα. Αυτό γίνεται χρησιμοποιώντας μια τεχνική που ονομάζεται δυναμικός προγραμματισμός. Η τάξη μεγέθους της υπολογιστικής προσπάθειας για αυτούς τους γρήγορους αλγόριθμους είναι μικρότερη από εκατό βήματα για έναν σωλήνα με 20 χάντρες.

## Χώρα προέλευσης θέματος

Ταϊβάν

## 2024-PL-04 – Η διαδρομή του ρομπότ

Ηλικιακή ομάδα: Α' – Β' Γυμνασίου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Η διαδρομή του ρομπότ

Το ρομπότ του Βασίλη κινείται σε πλακίδια ακολουθώντας ένα σύνολο κανόνων:

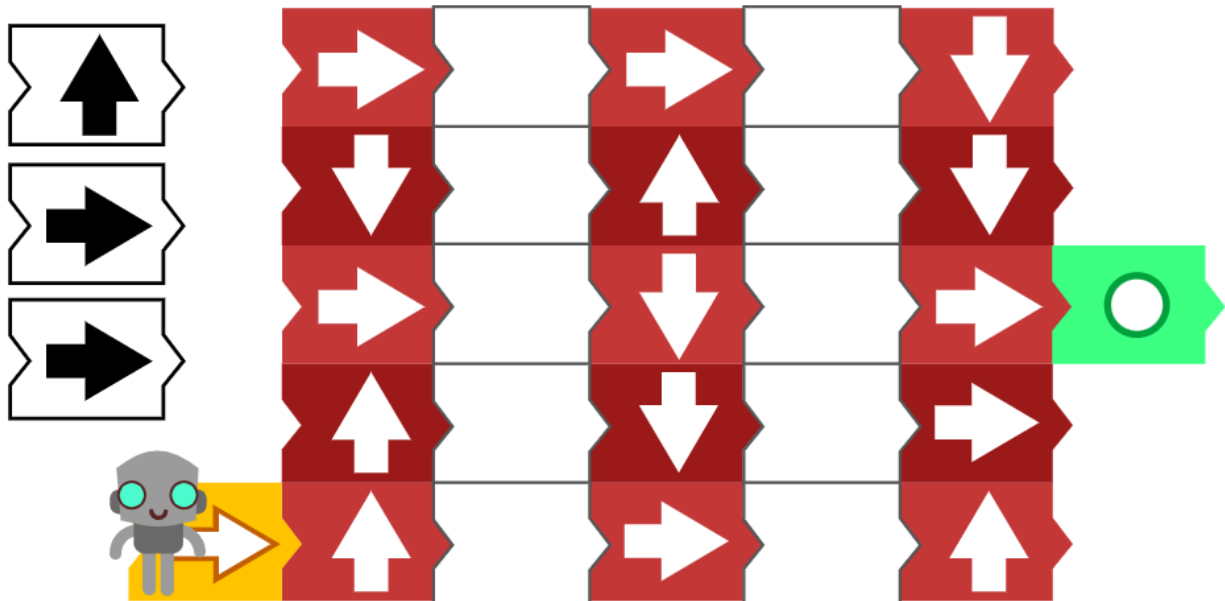
- Το ρομπότ ξεκινά από το πιο αριστερό πλακίδιο, όπως βλέπετε στην παρακάτω εικόνα.
- Το ρομπότ κινείται προς την κατεύθυνση του βέλους στο οποίο στέκεται.
- Εάν το ρομπότ απομακρυνθεί από τα πλακίδια, σταματά.

Ορισμένα πλακίδια είναι προκαθορισμένα όπως φαίνεται παρακάτω. Στον Βασίλη δίνονται τρία επιπλέον πλακίδια.

### Ερώτηση

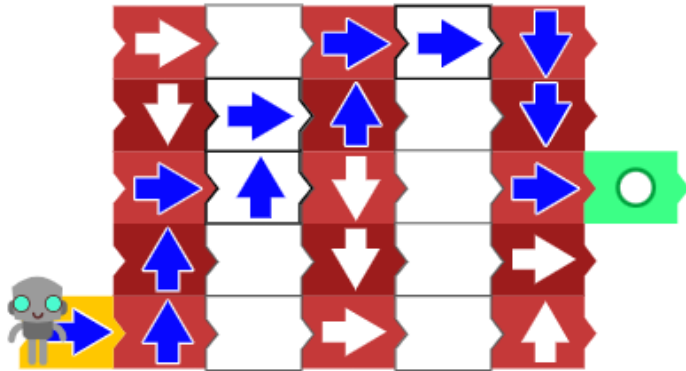
Βοηθήστε τον Βασίλη να τοποθετήσει και τα τρία επιπλέον πλακίδια στις κενές περιοχές, έτσι ώστε το ρομπότ του να φτάσει στο πράσινο πλακίδιο με τον κύκλο.

*Σύρτε τα βέλη στη σωστή θέση. Κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.*



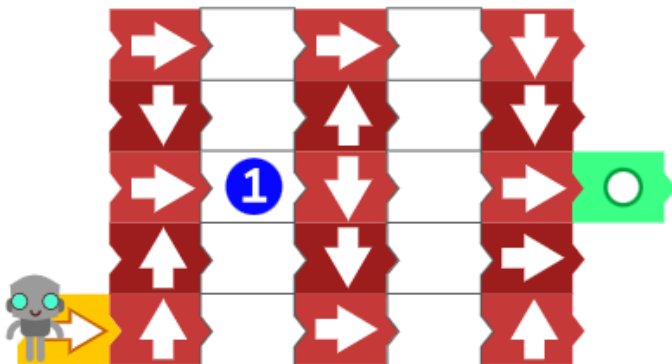
## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Όπως έχει ήδη υποδειχθεί, το ρομπότ του Βασίλη θα κινηθεί πρώτα προς τα δεξιά, δύο φορές προς τα πάνω και πάλι προς τα δεξιά. Αυτό σημαίνει ότι μια κάρτα πρέπει να τοποθετηθεί στη θέση που σημειώνεται με το 1:



Οι δύο τύποι διαθέσιμων καρτών είναι «πάνω» ή «δεξιά». Αν επιλέξετε «δεξιά», το ρομπότ θα κινηθεί προς τα δεξιά, μετά δύο φορές προς τα κάτω και πάλι προς τα δεξιά. Τώρα θα μπορούσατε είτε να επιλέξετε πάλι «δεξιά» είτε «πάνω» και μετά «δεξιά». Και στις δύο περιπτώσεις, το ρομπότ θα προσγειωνόταν κάτω από τον επιθυμητό στόχο:





επιστρέψετε και θα δοκιμάσετε έναν άλλο τρόπο.

Στην επιστήμη των υπολογιστών, αυτό ονομάζεται αναζήτηση σε βάθος (depth-first search), επειδή πρώτα δοκιμάζεται εντελώς ένας δρόμος και μετά δοκιμάζεται ο επόμενος. Επειδή συχνά πρέπει να πάτε ένα ή περισσότερα βήματα πίσω για να δοκιμάσετε μια διαφορετική λύση, κάνετε backtracking. Σε κάθε περίπτωση, όμως, λύνετε το πρόβλημα με την προς τα εμπρός ανίχνευση.

Ωστόσο, ορισμένα προβλήματα επιλύονται καλύτερα προς τα πίσω, κοιτάζοντας από τον στόχο για να δείτε πώς θα μπορούσατε να φτάσετε εκεί. Στην πραγματικότητα, αυτό το είδος προβλήματος δεν είναι κατάλληλο για να λυθεί προς τα πίσω, διότι ενώ όταν πηγαίνετε προς τα εμπρός ξέρετε πάντα ποια κάρτα είναι η επόμενη, όταν πηγαίνετε προς τα πίσω μπορεί να σημαίνει ότι έρχεστε από πολλές κάρτες (το βέλος δείχνει προς τα εμπρός και πηγαίνοντας προς τα πίσω πρέπει να ψάξετε όλες τις κάρτες γύρω του για να δείτε αν δείχνουν προς την κατεύθυνση της κάρτας).

Αλλά στη συγκεκριμένη περίπτωση, εξακολουθεί να είναι εύκολο να δείτε ποια λύση μπορεί να είναι η μόνη σωστή.

Υπολογιστική σκέψη

Ένας υπολογιστής είναι καλός στο να δοκιμάζει απλώς όλες τις πιθανές περιπτώσεις. Δεδομένου ότι υπάρχουν μόνο (για έναν υπολογιστή) πολύ λίγες πιθανές θέσεις και πολύ λίγες πιθανές κάρτες διαθέσιμες, ένας υπολογιστής μπορεί πολύ γρήγορα να δοκιμάσει απλά όλες τις πιθανότητες. Ένας τέτοιος σχετικά «αφελής» αλγόριθμος θα έλυne καλά το πρόβλημα.

Ωστόσο, οι άνθρωποι χρειάζονται λίγο περισσότερο χρόνο για να δοκιμάσουν τα πάντα. Επομένως, αξίζει να αναλύσουμε το πρόβλημα πιο προσεκτικά και να διατυπώσουμε περιορισμούς που μειώνουν κατά πολύ τον αριθμό των περιπτώσεων που πρέπει να δοκιμαστούν. Για παράδειγμα, μια κάρτα πρέπει να βρίσκεται στην τρίτη υψηλότερη θέση της αριστερής κόκκινης στήλης. Αυτό περιορίζει σημαντικά τον χώρο αναζήτησης πιθανών λύσεων, έτσι ώστε τελικά να χρειάζεται να δοκιμαστούν μόνο δύο παραλλαγές, όπως φαίνεται στην επεξήγηση της απάντησης.

Η υπολογιστική σκέψη δεν σημαίνει ότι απλώς δοκιμάζουμε όλες τις πιθανές περιπτώσεις όπως κάνει ένας υπολογιστής, αλλά ότι λαμβάνουμε υπόψη απλουστεύσεις που χρησιμοποιούν και οι επιστήμονες της πληροφορικής. Σε αυτή την περίπτωση, αυτό θα μπορούσε να είναι μια εφαρμοσμένη αναζήτηση σε βάθος με οπισθοδρόμηση ή μια αναζήτηση προς τα πίσω αντί για αναζήτηση προς τα εμπρός.

## Χώρα προέλευσης θέματος

Πολωνία

## 2024-CH-03a – Κάρτες Ricca

Ηλικιακή ομάδα: Α' – Β' Γυμνασίου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Κάρτες Ricca

Η Βαρβάρα συλλέγει κάρτες με τέρατα, τα «Riccas». Εδώ είναι οι κάρτες Ricca:



Μια κάρτα δείχνει τις ιδιότητες ενός Ricca, για παράδειγμα το όνομα (👤) ή αν το Ricca έχει δόντια (🦷). Οι ιδιότητες έχουν τιμές. Στην κάρτα 2, για παράδειγμα, (👤) έχει την τιμή MONI, και (🦷) έχει την τιμή ✓. Το Ricca στην κάρτα ονομάζεται επομένως MONI και έχει δόντια.

Η Βαρβάρα συνειδητοποιεί ότι υπάρχουν μόνο τρεις διαφορετικοί τύποι τιμών στις κάρτες, που φαίνονται παρακάτω:

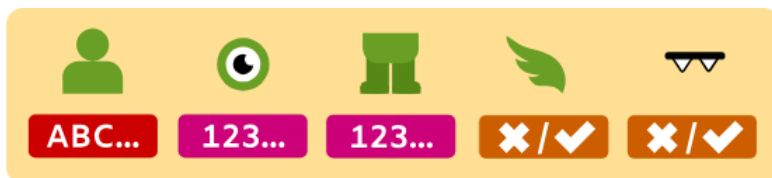
<b>ABC...</b>	Κείμενο, δηλαδή πολλά γράμματα στην σειρά
<b>123...</b>	Αριθμοί, όπως 1, 2, 3...
<b>✗ / ✓</b>	Οι χαρακτήρες ✗ και ✓, που σημαίνουν "Ναι" και "Όχι".

## Ερώτηση

Αναθέστε τις ιδιότητες στους σωστούς τύπους.  
Σύρετε τους τύπους τιμών στα πλαίσια κάτω από τις ιδιότητες. Κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.

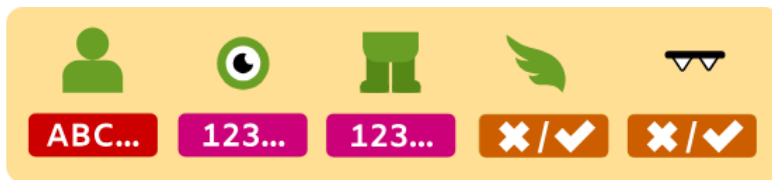
## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Η σωστή απάντηση είναι:



Ο ακόλουθος πίνακας αντιστοιχίζει τους τύπους δεδομένων στις τιμές τους για κάθε κάρτα.

Card					
1	JOSI	3	2	no	yes
2	MONI	3	2	no	yes
3	KILI	2	2	yes	yes
4	BENI	2	2	no	yes
5	LORI	2	5	no	yes
6	PHIL	2	2	no	yes

Από τον πίνακα διαβάζουμε ότι:

- Όλες οι τιμές της ιδιότητας  είναι ο τύπος ABC.
- Όλες οι τιμές της ιδιότητας  και  είναι ο τύπος 123
- Όλες οι τιμές της ιδιότητας  και  είναι τύπου ΝΑΙ/ΟΧΙ

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Η Βαρβάρα ανακάλυψε ότι στις κάρτες της Ricca υπάρχουν διαφορετικοί τύποι τιμών. Η διάκριση μεταξύ αυτών των τύπων μπορεί να είναι χρήσιμη κατά την οργάνωση των καρτών. Για παράδειγμα, επειδή η Βαρβάρα γνωρίζει ότι όλες οι τιμές της ιδιότητας είναι αριθμοί, μπορεί να συγκρίνει τον αριθμό των ποδιών σε κάθε Ricca και να ταξινομήσει τις κάρτες με βάση τον αριθμό των ποδιών.

Μια κάρτα Ricca είναι σαν ένα σύνολο δεδομένων που αποθηκεύεται σε έναν υπολογιστή. Η ηλεκτρονική μνήμη του υπολογιστή μπορεί να αποθηκεύσει μόνο 1 και 0. Επομένως, τα προγράμματα υπολογιστών χρησιμοποιούν «τύπους τιμών», οι οποίοι στην επιστήμη των υπολογιστών ονομάζονται τύποι δεδομένων, για να καθορίσουν τον τρόπο με τον οποίο ο υπολογιστής πρέπει να ερμηνεύσει και να χειριστεί αυτά τα 1 και 0. Επιπλέον, με τη χρήση τύπων δεδομένων, είναι δυνατόν να ελέγχεται αν και πώς μπορούν να συνδυαστούν τα αποθηκευμένα δεδομένα.

Οι τύποι τιμών αυτής της εργασίας Bebras αντιστοιχούν σε σημαντικούς τύπους δεδομένων από τον προγραμματισμό:

- Τα δεδομένα του πρώτου τύπου είναι ακολουθίες χαρακτήρων «string» (γράμματα, αριθμοί και άλλοι χαρακτήρες).
- Ο αριθμός αντιστοιχεί στον τύπο «integer», τα δεδομένα αυτού του τύπου είναι ακέραιοι αριθμοί, δηλαδή ...-2, -1, 0, 1, 2.
- yes/no αντιστοιχεί στον τύπο «Boolean» που γνωρίζει μόνο δύο διαφορετικές τιμές οι οποίες συχνά ονομάζονται «True» και «False».

### Υπολογιστική σκέψη

**Αφαίρεση:** Οι συμμετέχοντες πρέπει να φιλτράρουν τις σχετικές πληροφορίες (ιδιότητες) και να αντιστοιχίσουν έναν τύπο δεδομένων στις τιμές κάθε ιδιότητας. Κατά τη δημιουργία ενός πίνακα ή μιας βάσης δεδομένων, είναι σημαντικό να επιλέγονται οι σωστοί τύποι δεδομένων, ώστε να είναι δυνατή η διήθηση και η αναζήτηση των δεδομένων που συλλέγονται. Η δημιουργία μιας καλά δομημένης βάσης δεδομένων είναι μια πολύ σημαντική δεξιότητα που απαιτείται κατά τη δημιουργία ενός συστήματος για τη διαχείριση της ροής παραγωγής σε ένα εργοστάσιο ή του καταλόγου ειδών για ένα κατάστημα.

Χώρα προέλευσης θέματος

Ελβετία

## 2024-DE-04a – Ηλιόλουστες μέρες

Ηλικιακή ομάδα: Α' – Β' Γυμνασίου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Ηλιόλουστες μέρες



Ο Πασχάλης λέει: «Τις ηλιόλουστες ημέρες υπάρχει τουλάχιστον ένας κάστορας που κολυμπάει σε κάθε λίμνη».

Η Μαρίνα απαντά: «Αυτό δεν είναι αλήθεια. Για παράδειγμα, την περασμένη Κυριακή δεν ήταν έτσι».

### Ερώτηση

Ας υποθέσουμε ότι η Μαρίνα έχει δίκιο. Τι συνέβη την περασμένη Κυριακή; Συμπληρώστε το κενό αναλόγως. Σύρετε το σωστό κείμενο στο κενό και κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.

Την προηγούμενη Κυριακή είχε ήλιο και

όλες οι λίμνες ήταν γεμάτες από κάστορες που κολυμπύσαν.

κανένας κάστορας δεν κολύπησε στην λίμνη με τον καταρράκτη.

ο κάστορας Θωμάς κολύπησε σε όλες τις λίμνες.

ο κάστορας Θωμάς δεν κολύπησε καθόλου.

## Απάντηση

Η σωστή απάντηση είναι:

Την προηγούμενη Κυριακή είχε ήλιο και

κανένας κάστορας δεν κολύμπησε στην λίμνη με τον καταρράκτη.

## Εξήγηση απάντησης

Η λύση είναι:

Την προηγούμενη Κυριακή είχε ήλιο και

κανένας κάστορας δεν κολύμπησε στην λίμνη με τον καταρράκτη.

Σε αυτή την περίπτωση, δεν έχει κάθε λίμνη έναν κάστορα που κολυμπάει μέσα της (αυτή με τον παγωμένο καταρράκτη δεν έχει). Επίσης, γνωρίζουμε ότι την περασμένη Κυριακή είχε ηλιοφάνεια. Έτσι, η περασμένη Κυριακή είναι ένα παράδειγμα ηλιόλουστης ημέρας που έχει τουλάχιστον μία λίμνη χωρίς κάστορες που κολυμπούν. Επομένως, ο Πασχάλης κάνει λάθος όταν λέει ότι σε κάθε ηλιόλουστη μέρα, κάθε λίμνη έχει έναν κάστορα να κολυμπάει μέσα της.

Οι άλλες επιλογές απαντήσεων δεν αποδεικνύουν ότι ο Πασχάλης έχει άδικο:

Αυτό σημαίνει ότι σε κάθε λίμνη υπάρχουν κάστορες που κολυμπούν μέσα της, πράγμα που ταιριάζει με τη δήλωση του Πασχάλη.

Ο κάστορας Θωμάς κολυμπούσε σε κάθε λίμνη - αυτό σημαίνει ότι σε κάθε λίμνη κολυμπάει τουλάχιστον ο Θωμάς, ταιριάζοντας και πάλι με τη δήλωση του Πασχάλη.

Ο κάστορας Θωμάς δεν κολύμπησε καθόλου - αλλά θα μπορούσαν να υπάρχουν και άλλοι κάστορες που κολυμπούν σε όλες τις λίμνες, οπότε η δήλωση του Πασχάλη εξακολουθεί να ικανοποιείται.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής



Ο Πασχάλης και η Μαρίνα συζητούν αν η δήλωση του Πασχάλη είναι αληθής ή ψευδής. Η θεωρία των δηλώσεων που είναι είτε αληθείς είτε ψευδείς ονομάζεται λογική. Πολύπλοκες δηλώσεις μπορούν να γίνουν χρησιμοποιώντας τελεστές όπως AND, OR, NOT και ποσοδείκτες όπως EVERY και AT LEAST ONE. Η λογική βοηθά στην αποφυγή παρεξηγήσεων και λανθασμένων συμπερασμάτων σε συζητήσεις. Επιπλέον, η λογική αποτελεί το θεμέλιο του τρόπου λειτουργίας των υπολογιστών. Κατ' αρχάς, κάθε πρόγραμμα υπολογιστή εκτελείται βασικά σε μικροσκοπικά φυσικά δομικά στοιχεία που αναπαριστούν λογικούς τελεστές όπως το NOT και το AND. Δεύτερον, οι γλώσσες που χρησιμοποιούνται για να δοθούν οδηγίες σε έναν υπολογιστή χρησιμοποιούν λογικές προτάσεις, π.χ. «ΑΝ είναι καλοκαίρι ΚΑΙ ο ήλιος λάμπει, ΤΟΤΕ ενεργοποίησε τον κλιματισμό».

### Υπολογιστική σκέψη

Επειδή οι υπολογιστές βασίζονται στη λογική, η εργασία σε αυτούς απαιτεί συχνά την ενασχόληση με τη λογική. Για παράδειγμα, ένας προγραμματιστής μπορεί να πρέπει να μεταφράσει μια κατάσταση από τον πραγματικό κόσμο σε μια λογική δήλωση για να την κοινοποιήσει στον υπολογιστή. Ή μπορεί να χρειαστεί να βρει το ακριβές αντίθετο μιας συνθήκης για να βεβαιωθεί ότι το πρόγραμμά του περιλαμβάνει οδηγίες για όλες τις πιθανές περιπτώσεις. Αυτές οι εργασίες απαιτούν δεξιότητες υπολογιστικής σκέψης με τη μορφή της αποσύνθεσης. Υπάρχει ακόμη και ο όρος «λογική σκέψη» που είναι στενά συνδεδεμένος με την «υπολογιστική σκέψη».

## Χώρα προέλευσης θέματος



Γερμανία

## 2024-BG-01b – Μπάλες

Ηλικιακή ομάδα: Α' – Β' Γυμνασίου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Μπάλες

Παρακάτω βλέπετε μια ακολουθία από κόκκινες  και μπλε  μπάλες:



Για κάθε μπάλα, μετράμε τώρα πόσες μπλε μπάλες βρίσκονται στα δεξιά της (συμπεριλαμβανομένης της ίδιας της μπάλας) και το σημειώνουμε ως εξής:

3, 3, 2, 1, 1, 1.

Στο επόμενο βήμα, σημειώνουμε ξανά την ίδια ακολουθία. Αλλά αν ο αριθμός είναι ζυγός (άρτιος), σημειώνουμε ένα 0, αν είναι μονός (περιττός), τότε ένα 1. Αυτό οδηγεί σε αυτή την ακολουθία:

110111

### Ερώτηση

Δίνεται η παρακάτω ακολουθία των αριθμών 0 και 1:

01110100

Κατασκευάστε μια ακολουθία μπλε και κόκκινων μπάλων που να ταιριάζει με αυτή την αναπαράσταση.



## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

Υπάρχουν διάφοροι τρόποι για να βρείτε αυτή την απάντηση. Ένας τρόπος είναι να περάσετε την ακολουθία των αριθμών από τα δεξιά προς τα αριστερά και να συμπεράνετε λογικά αν η τρέχουσα μπάλα πρέπει να είναι κόκκινη ή μπλε.

Ας δούμε τις μπάλες με τη σειρά:

1. Ας κοιτάξουμε την πρώτη μπάλα στα δεξιά. Παίρνουμε το 0 σε αυτό το σημείο ως ένδειξη ότι η μπάλα πρέπει να είναι κόκκινη, επειδή μόνο μια κόκκινη μπάλα διατηρεί το άθροισμα όλων των μπλε μπαλών ίσο.

0	1	1	1	0	1	0	0
							Κόκκινη

2. Στη δεύτερη θέση από τα δεξιά βρίσκουμε ένα 0. Το επιχείρημα παραμένει το ίδιο όπως και πριν: μόνο με μια κόκκινη μπάλα το άθροισμα όλων των μπλε μπαλών παραμένει σωστό και συμπεριλαμβανομένης αυτής της θέσης αυτή τη στιγμή.

0	1	1	1	0	1	0	0
						Κόκκινη	Κόκκινη

3. Στην τρίτη θέση από τα δεξιά βρίσκουμε ένα 1, το οποίο υποδηλώνει ότι ο αριθμός των μπλε μπαλών θα πρέπει να είναι μονός από αυτό το σημείο και μετά. Εφόσον γνωρίζουμε ήδη ότι οι δύο μπάλες στα δεξιά είναι κόκκινες, άρα η τρέχουσα μπάλα πρέπει να είναι μπλε για να μετακινηθούμε από έναν ευθύγραμμο σε έναν μονό αριθμό μπλε μπαλών.

0	1	1	1	0	1	0	0
					Μπλε	Κόκκινη	Κόκκινη

4. Στην τέταρτη θέση από δεξιά βρίσκουμε ένα 0. Για να αντικαταστήσουμε τις προηγούμενες μονές μπάλες (2x κόκκινες και 1x μπλε) για να πάρουμε έναν ζυγό αριθμό μπλε μπαλών, ο αριθμός των μπλε μπαλών της τέταρτης μπάλας πρέπει να είναι μπλε.

0	1	1	1	0	1	0	0
				Μπλε	Μπλε	Κόκκινη	Κόκκινη

5. Στην πέμπτη θέση από δεξιά βρίσκουμε ένα 1. Αφού το προηγούμενο άθροισμα των μπλε μπαλών ήταν ζυγό και τώρα θα είναι μονό, η νέα μπάλα πρέπει να είναι επίσης μπλε.

0	1	1	1	0	1	0	0
			Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

6. Στην έκτη θέση από δεξιά βρίσκεται ο αριθμός 1. Εφόσον ο προηγούμενος αριθμός μπλε μπαλών ήταν περιττός, η τρέχουσα μπάλα πρέπει να είναι κόκκινη, ώστε ο αριθμός που εμφανίζεται να είναι περιττός.

0	1	1	1	0	1	0	0
		Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

7. Στην έβδομη θέση από τα δεξιά βρίσκουμε άλλο ένα 1. Η τρέχουσα ακολουθία περιέχει ήδη έναν περιττό αριθμό μπλε μπαλών- για να διατηρηθεί αυτή η κατάσταση, η νέα μπάλα πρέπει να έχει κόκκινο χρώμα.

0	1	1	1	0	1	0	0
	Κόκκινη	Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

8. Στην όγδοη θέση από τα δεξιά (πρώτη θέση από τα αριστερά) βρίσκουμε τον αριθμό 0. Για να γίνει και πάλι ζυγός ο μονός αριθμός μπλε μπαλών, πρέπει να είναι μια μπλε μπάλα.

0	1	1	1	0	1	0	0
Μπλε	Κόκκινη	Κόκκινη	Μπλε	Μπλε	Μπλε	Κόκκινη	Κόκκινη

Συνοπτικά, σε κάθε μετάβαση από το 0 στο 1 ή από το 1 στο 0 πρέπει να χρησιμοποιείται μια μπλε μπάλα, ενώ μια συνεχής ακολουθία των ίδιων αριθμών υποδεικνύει κόκκινες μπάλες. Αυτές δεν επηρεάζουν το άθροισμα των μπλε μπαλών:

0	1	1	1	0	1	0	0
4 Μπλε	3	3	3 Μπλε	2 Μπλε	1 Μπλε	0	0
	Κόκκινες	Κόκκινες				Κόκκινες	Κόκκινες

Τότε η ακολουθία των μπλε μπαλών είναι 43332100.



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

- Η κωδικοποίηση χρησιμοποιείται για την αναπαράσταση πολύπλοκων θεμάτων με τρόπο ώστε οι υπολογιστές να μπορούν εύκολα να τα επεξεργαστούν. Συχνά άσχετα δεδομένα μπορούν να παραλειφθούν χωρίς να επηρεαστούν τα κεντρικά στοιχεία της πληροφορίας. Αυτή η εργασία προτείνει μια κωδικοποίηση του τρόπου με τον οποίο μπορεί να αποθηκευτεί ο ακριβής αριθμός και η θέση των μπλε και κόκκινων μπαλών χωρίς τα ίδια τα πραγματικά χρώματα. Λόγω της επιλεγμένης κωδικοποίησης βρισκόμαστε στη θέση της αρχικής πληροφορίας (τα χρώματα και η θέση των αρχικών μπαλών).
- Υπό ορισμένες συνθήκες, η κωδικοποίηση μπορεί επίσης να εξοικονομήσει αποθηκευτικό χώρο, κάτι που ονομάζουμε συμπίεση. Για την αναπαράσταση δυαδικής πληροφορίας (όπως στην πληροφορία αν ένας αριθμός είναι ζυγός ή περιττός) αρκεί ένα μόνο bit πληροφορίας, ενώ για την αποθήκευση οποιουδήποτε αριθμού ή χρώματος απαιτείται περισσότερος αποθηκευτικός χώρος.
- Ως πρώτο ενδιάμεσο βήμα, προσδιορίστηκε μια αθροιστική συχνότητα για τον υπολογισμό του αθροίσματος των στοιχείων που έχουν ένα συγκεκριμένο χαρακτηριστικό (εδώ συγκεκριμένα τον αριθμό των μπλε μπαλών από την τρέχουσα θέση). Η αθροιστική συχνότητα είναι ένα εργαλείο της στατιστικής, το οποίο χαρακτηρίζεται ιδιαίτερα από το γεγονός ότι η πληροφορία που προκύπτει συσσωρεύεται (εδώ όχι μόνο μια ένδειξη για το χρώμα της τρέχουσας μπάλας, αλλά και για τα χρώματα των μπαλών πίσω από αυτήν). Οι λύσεις προβλημάτων που χρησιμοποιούν τη σωρευτική συχνότητα συχνά επωφελούνται από το γεγονός ότι μπορείτε να επεξεργαστείτε πολλές αιτήσεις γρήγορα χωρίς να χρειάζεται να έχετε πρόσβαση στα αρχικά δεδομένα.

### Υπολογιστική σκέψη

Κατά τη διάρκεια της διαδικασίας κωδικοποίησης ενός προβλήματος, οι μαθητές εξασκούν την ικανότητα της αφαίρεσης: παραλείπονται οι άσχετες πληροφορίες και γίνεται σταδιακή μείωση των λεπτομερειών, χωρίς να χάνεται το κεντρικό πληροφοριακό περιεχόμενο.

## Χώρα προέλευσης θέματος

Βουλγαρία

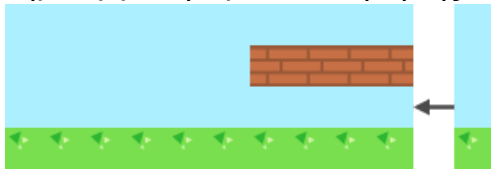
## 2024-DE-07 – Ηλεκτρονικό παιχνίδι

Ηλικιακή ομάδα: Α' - Β' Γυμνασίου

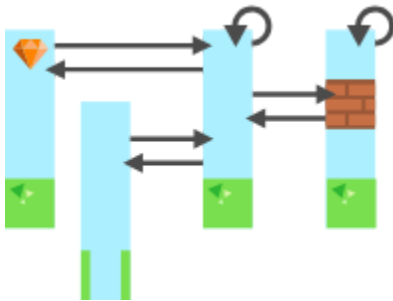
Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Ηλεκτρονικό παιχνίδι

Σε ένα ηλεκτρονικό παιχνίδι, το φόντο αποτελείται από μια ακολουθία πλακιδίων. Ο υπολογιστής προσθέτει συνεχώς ένα νέο πλακίδιο στα δεξιά της ακολουθίας και ταυτόχρονα αφαιρεί ένα πλακίδιο από τα αριστερά. Με αυτόν τον τρόπο, ο υπολογιστής δημιουργεί την ψευδαίσθηση της κίνησης.



Ο υπολογιστής επιλέγει ένα νέο πλακίδιο προς προσθήκη χρησιμοποιώντας το διάγραμμα στα δεξιά. Κοιτάζει το προηγούμενο πλακίδιο και ελέγχει τα βέλη που προέρχονται από αυτό. Στη συνέχεια επιλέγει τυχαία ένα από τα πλακίδια στο οποίο δείχνει ένα από τα βέλη.

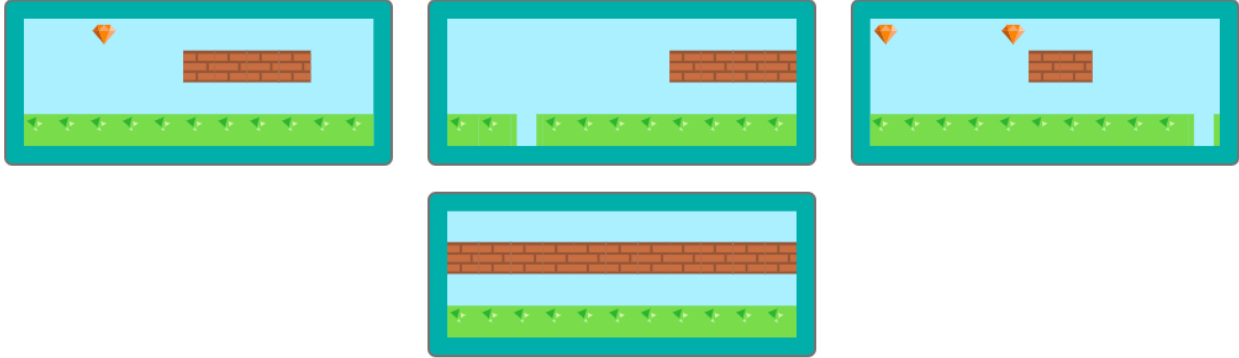


Για παράδειγμα, μετά το  πλακίδιο, ο υπολογιστής μπορεί να διαλέξει είτε το  ή το

 πλακίδιο.

## Ερώτηση

Μία από τις παρακάτω εικόνες ΔΕΝ αποτελεί έγκυρο φόντο σε αυτό το παιχνίδι. Ποια από αυτές είναι;

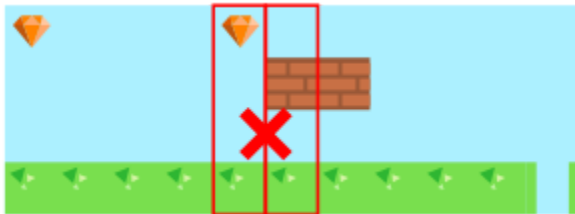



## Απάντηση

Απάντηση 3

## Εξήγηση απάντησης

Το φόντο που φαίνεται στην ακόλουθη απάντηση δεν έχει δημιουργηθεί σύμφωνα με το διάγραμμα.



Ένα πλακίδιο με διαμάντι μπορεί να ακολουθήσει μόνο το  πλακίδιο, κάτι που δεν συμβαίνει εδώ.

Υπάρχουν διάφοροι τρόποι για να βρείτε τη λύση σε αυτό το ερώτημα. Μια απλή στρατηγική είναι να ελέγξετε οποιαδήποτε δεδομένη εικόνα φόντου και να συγκρίνετε κάθε πλακίδιο με το διάγραμμα για να δείτε αν είναι έγκυρο. Ο πιο γρήγορος τρόπος είναι αυτός:



- Ελέγξτε τα πλακίδια στο διάγραμμα, ξεκινώντας από τα πλακίδια που έχουν μόνο ένα βέλος. Αυτά τα βέλη μπορούν να θεωρηθούν ως περιορισμοί.  $A \rightarrow B$  σημαίνει «το πλακίδιο A πρέπει να ακολουθείται από το πλακίδιο B».

- Στη συνέχεια ελέγξτε εάν αυτός ο περιορισμός ισχύει για κάθε εικόνα φόντου.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Ορισμένα παιχνίδια στον υπολογιστή – όπως τα ατελείωτα παιχνίδια δρομέων – έχουν φόντο που κινείται οριζόντια, δημιουργώντας την ψευδαίσθηση ότι ο παίκτης κινείται σε έναν κόσμο φαντασίας. Μερικές φορές το φόντο δεν είναι μια σταθερή εικόνα, αλλά δημιουργείται αυτόματα από τον υπολογιστή. Αυτό ονομάζεται procedural generation. Συνήθως, μικρότερα στοιχεία συνδυάζονται τυχαία για να δημιουργήσουν μια μεγάλη ποικιλία από φόντο.

Ωστόσο, ο συνδυασμός στοιχείων δεν μπορεί να είναι εντελώς τυχαίος, αλλά πρέπει να ακολουθεί ορισμένους κανόνες.

Στην ερώτηση, αυτοί οι κανόνες αντιπροσωπεύονται από ένα διάγραμμα πλακιδίων και τέτοια διαγράμματα, όπου τα στοιχεία συνδέονται με βέλη, ονομάζονται κατευθυνόμενα γραφήματα. Τα στοιχεία των διαγραμμάτων ονομάζονται κόμβοι και τα βέλη είναι κατευθυνόμενες άκρες. Τα κατευθυνόμενα γραφήματα χρησιμοποιούνται για όλους τους τύπους μοντελοποίησης.

Τα πεπερασμένα αυτόματα πηγαίνουν ένα βήμα παραπέρα: Έχουν καταστάσεις, κατάσταση έναρξης, καταστάσεις λήξης και κανόνες για το πώς να μετακινηθείτε από τη μια κατάσταση στην άλλη. Τα πεπερασμένα αυτόματα χρησιμοποιούνται συχνά για μοντελοποίηση στην επιστήμη των υπολογιστών.

### Υπολογιστική σκέψη

Για να λύσετε την άσκηση πρέπει να διαβάσετε και να χρησιμοποιήσετε το κατευθυνόμενο γράφημα. Το γράφημα καθορίζει τη σύνθεση της εικόνας φόντου. Μπορεί να θεωρηθεί ως μια αφαίρεση του μηχανισμού που δημιουργεί αυτόματα εικόνες.

Επιπλέον, η αναγνώριση μοτίβων μπορεί να χρησιμοποιηθεί για να χωρίσει οπτικά το φόντο σε μια σειρά πλακιδίων και να ανιχνεύσει ελαττωματικές μεταβάσεις. Επομένως, αυτή η άσκηση περιλαμβάνει αλγοριθμική σκέψη, αφαίρεση και αναγνώριση προτύπων.

Χώρα προέλευσης θέματος

Γερμανία




# 2024-DE-08 – Ξενάγηση δέντρων

Ηλικιακή ομάδα: Α' – Β' Γυμνασίου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ





Θέμα : Ξενάγηση δέντρων










Η Αναστασία προσφέρει ξεναγήσεις στο δάσος και μιλάει για τα δέντρα που συναντά. Από προηγούμενες ξεναγήσεις, γνωρίζει ποια δέντρα είναι ιδιαίτερα δημοφιλή στους επισκέπτες.

Ξενάγηση 1	
Ξενάγηση 2	
Ξενάγηση 3	

Εδώ το Δέντρο 1 < Δέντρο 2 σημαίνει ότι το Δέντρο 1 είναι λιγότερο δημοφιλές από το Δέντρο 2.

Στην επόμενη ξεναγηση, η Αναστασία θέλει να δείξει τα πιο δημοφιλή δέντρα στο τέλος. Αποφασίζει να βρει μια καλή σειρά που να μην έρχεται σε σύγκρουση με τον παραπάνω πίνακα.

Παράδειγμα: Εάν η Αναστασία σκοπεύει να παρουσιάσει τα δέντρα  και  στη σωστή σειρά, τότε πρέπει να δείξει το  πριν από το .

Δείχνοντας το  πριν από το  θα ερχόταν σε σύγκρουση με την Ξενάγηση 1, όπου το  ήταν πιο δημοφιλές από το . Στη νέα ξεναγηση, η Αναστασία θα ήθελε να δείξει τα δέντρα , , ,  και .

## Ερώτηση

Βοηθήστε την Αναστασία να οργανώσει τη νέα ξενάγηση για να επισκεφθεί αυτά τα δέντρα με τη σειρά δημοτικότητας.

Σύρετε τα δέντρα στις σωστές τελείες. Κάντε κλικ στο κουμπί "Αποθήκευση" όταν τελειώσετε.

## Απάντηση

Η σωστή απάντηση είναι:



## Εξήγηση απάντησης

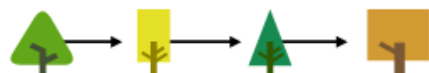
Η σωστή απάντηση είναι:




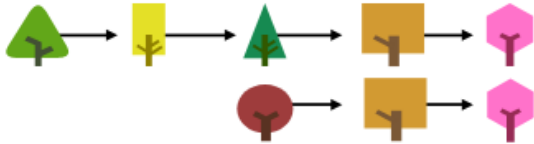
Μια κατάλληλη σειρά δεν πρέπει να έρχεται σε αντίθεση με τη δημοτικότητα των δέντρων από προηγούμενες περιηγήσεις:

Ξενάγηση 1	
Ξενάγηση 2	
Ξενάγηση 3	

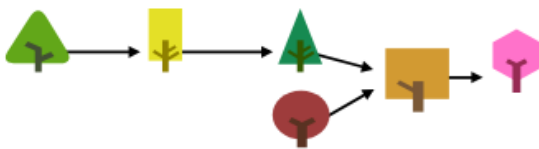
Θα προσπαθήσουμε τώρα να συνδυάσουμε τις τρεις περιηγήσεις και να παρουσιάσουμε τις περιηγήσεις ως διάγραμμα. Σε αυτό το διάγραμμα, δύο δέντρα συνδέονται πάντα με ένα βέλος όταν πρέπει να επισκεφθείτε ένα δέντρο πριν από ένα άλλο δέντρο. Εδώ είναι το διάγραμμα για την ξενάγηση 1:




Αυτές οι πληροφορίες μπορούν τώρα να συνδεθούν με την ξενάγηση 3. Αρχικά, διαπιστώνουμε ότι και η ξενάγηση 1 και η ξενάγηση 3 περιέχουν το δέντρο  :



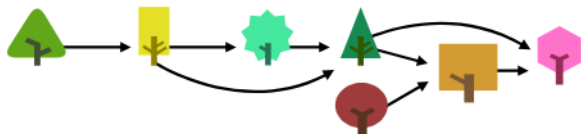
Χάρη σε αυτήν την επικάλυψη, μπορούμε επίσης να συνδυάσουμε τα δύο διαγράμματα σε ένα:



Τέλος, προσπαθούμε να ενσωματώσουμε την ξενάγηση 2 στο προηγούμενο διάγραμμα (παραπάνω). Διαπιστώνουμε ότι τρία από τα τέσσερα δέντρα προηγουμένως είχαν ήδη συμπεριληφθεί στο διάγραμμα. Μόνο το  προστέθηκε πρόσφατα στο διάγραμμα:



Και πάλι, μπορούμε να συνδυάσουμε και τα δύο διαγράμματα σε ένα, διατηρώντας μόνο ένα αντίγραφο κάθε δέντρου και να προσθέσουμε τα βέλη ανάλογα:



Χρησιμοποιώντας το διάγραμμα, μπορούμε πλέον να λύσουμε το πρόβλημα πολύ εύκολα: Επισκεπτόμαστε τα δέντρα του διαγράμματος από αριστερά προς τα δεξιά κατά μήκος των βελών και έτσι χτίζουμε μια σειρά από τα δέντρα που πρέπει να συμπεριληφθούν στην ξενάγηση. Σημειώστε επίσης ότι δεν πρέπει να συμπεριλαμβάνεται κάθε δέντρο στην περιήγηση.

Η νέα ξενάγηση επισκέπτεται τα δέντρα ως εξής:





## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Σε αυτό το πρόβλημα πρέπει να ταξινομήσετε ένα σύνολο δέντρων με αύξουσα σειρά δημοτικότητας.

Έχετε πληροφορίες για ορισμένα δέντρα που συγκρίνονται σε ζευγάρια ως προς τη δημοτικότητά τους.

Για άλλα ζεύγη δέντρων όπως σύγκριση των  και  δεν είναι άμεση σύγκριση.

Ωστόσο, αυτά τα δέντρα μπορούν επίσης να συγκριθούν. Δύο ιδιότητες αυτής της εργασίας είναι κεντρικής σημασίας:

- Η σύγκριση δημοτικότητας είναι μεταβατική. Δηλαδή, αν δέντρο 1 < δέντρο 2 και δέντρο 2 < δέντρο 3, τότε δέντρο 1 < δέντρο 3.
- Οι τρεις ταξινομήσεις που έχουν ήδη δοθεί δεν έρχονται σε αντίθεση μεταξύ τους. Για παράδειγμα, θα ήταν πρόβλημα εάν σε μία ξενάγηση είχαμε: δέντρο 1 < δέντρο 2 και ταυτόχρονα σε μία άλλη: δέντρο 2 < δέντρο 1.

Χάρη σε αυτές τις ιδιότητες, η εργασία καταλήγει σε μια σειρά με τη μαθηματική έννοια. Μπορεί αρχικά να φαίνεται ότι αυτή η σειρά είναι ελλιπής, καθώς δεν βλέπουμε όλα τα δέντρα να συγκρίνονται σε ζεύγη. Στην πραγματικότητα, τα δέντρα σε αυτό το πρόβλημα μπορούν να τακτοποιηθούν σε μια ταξινομημένη σειρά (που ονομάζεται τοπολογική ταξινόμηση). Υπάρχει ένας σχετικά απλός αλγόριθμος για τον προσδιορισμό μιας τέτοιας τοπολογικής ταξινόμησης. Ωστόσο, για να συμβεί αυτό, δεν πρέπει να υπάρχουν συγκρούσεις στις αρχικές συγκρίσεις.

Μετά από πιο προσεκτική εξέταση, διαπιστώνουμε μάλιστα ότι η δημοτικότητα των δέντρων παρέχει την πλήρη εικόνα της ταξινόμησης. Χάρη στη μεταβατικότητα για όλα τα ζεύγη δέντρων, λαμβάνουμε μια ενιαία σωστή λύση για την τοπολογική ταξινόμηση όλων των δέντρων.

## Χώρα προέλευσης θέματος

Γερμανία

# 2024-TW-03-UK – Το πιο μακρύ βραχιόλι

Ηλικιακή ομάδα: Α' - Β' Γυμνασίου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Το πιο μακρύ βραχιόλι

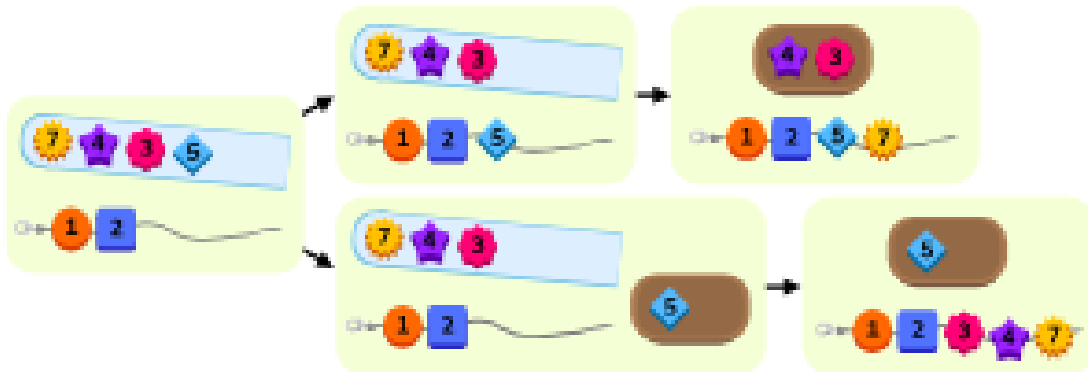
Ο Μάριος φτιάχνει ένα βραχιόλι. Παίρνει χάντρες με αριθμούς από ένα γυάλινο σωλήνα. Κολλάει μερικές χάντρες από αυτές και βάζει τις άλλες στο καφέ κουτί και δεν τις χρησιμοποιεί. Επιτρέπεται να περάσει μια χάντρα μόνο αν:

το κορδόνι είναι άδειο, ή

η χάντρα έχει μεγαλύτερο αριθμό από την τελευταία χάντρα στο κορδόνι.

Σε αυτό το παράδειγμα, η τελευταία χάντρα στο κορδόνι είναι η χάντρα 2. Στη συνέχεια, ο Μάριος μπορεί να περάσει τη χάντρα 5 από τον σωλήνα. Μπορεί επίσης να την αφήσει στο καφέ κουτί και να μην τη χρησιμοποιήσει.

Αν περάσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με τέσσερις χάντρες 1257. Αν δεν χρησιμοποιήσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με περισσότερες χάντρες 12347.



Ο Μάριος φτιάχνει ένα καινούργιο βραχιόλι από τις χάντρες αυτού του σωλήνα:



## Ερώτηση

Ποιος είναι ο μέγιστος αριθμός χαντρών που μπορεί να περάσει από το κορδόνι;

## Απάντηση

6

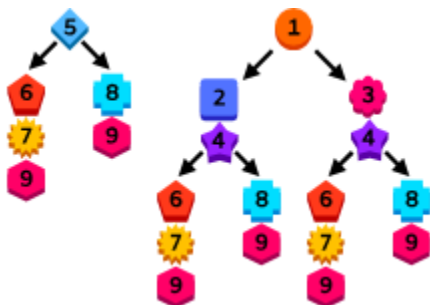
## Εξήγηση απάντησης

Η απάντηση είναι 6.

Θα μπορούσαμε να απαριθμήσουμε όλα τα βραχιόλια για αυτό το σωληνάριο και να δούμε ποια έχουν τον μεγαλύτερο αριθμό χαντρών. Αλλά αυτό θα ήταν χρονοβόρο. Ας ρίξουμε μια πιο προσεκτική ματιά στους αριθμούς:



Η πρώτη χάντρα είναι ο αριθμός 5. Όταν αυτή περνάει στο κορδόνι, οι επόμενες δυνατές χάντρες είναι μόνο οι 6, 7, 8 και 9, επομένως οι πιθανές ακολουθίες θα ήταν 5679 και 589. Εάν η χάντρα 5 αφηθεί στην άκρη και η επόμενη χάντρα, η χάντρα 1, μπει στο κορδόνι, υπάρχουν περισσότερες πιθανές ακολουθίες. Αυτό μπορεί να αναπαρασταθεί ξεκάθαρα σε ένα δέντρο αποφάσεων:



Εάν η χάντρα 1 αφηθεί επίσης στην άκρη και μία διαφορετική χάντρα περαστεί πρώτα, οι πιθανές αλληλουχίες περιέχονται ήδη σε μία από τις παραπάνω ακολουθίες. Εάν ξεκινήσετε με την χάντρα 2, για παράδειγμα, θα λάβετε 24679 ή 2489, που περιέχονται στις μεγαλύτερες ακολουθίες 124679 και 12489 αντίστοιχα.



Τα βραχιόλια με τον μεγαλύτερο αριθμό χαντρών ξεκινούν με τη χάντρα 1 και αποτελούνται από 6 χάντρες: 124679 ή 123679.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Κάθε σωλήνας περιέχει μια ακολουθία αριθμημένων χαντρών. Εάν ο Μάριος ακολουθήσει τον δεύτερο κανόνα (μόνο οι χάντρες στο κορδόνι που έχουν μεγαλύτερο αριθμό από την τελευταία), θα πάρει ένα βραχιόλι με αύξουσα ακολουθία. Για να κατασκευαστεί ένα βραχιόλι με όσο το δυνατόν περισσότερες χάντρες, πρέπει να καθορίσει η μεγαλύτερη ανοδική ακολουθία.

Για μεγάλες ακολουθίες, θα χρειαζόταν πολύς χρόνος για να σχηματιστούν όλες οι πιθανές αύξουσες υποακολουθίες και στη συνέχεια να προσδιοριστεί η μεγαλύτερη. Εάν ο σωλήνας περιείχε 20 χάντρες, για παράδειγμα, η υπολογιστική προσπάθεια θα ήταν της τάξης ενός εκατομμυρίου βημάτων.

Ευτυχώς, στην επιστήμη των υπολογιστών έχουν αναπτυχθεί αλγόριθμοι που μπορούν να βρουν τη μεγαλύτερη αύξουσα υποακολουθία πιο γρήγορα. Αυτό γίνεται χρησιμοποιώντας μια τεχνική που ονομάζεται δυναμικός προγραμματισμός. Η τάξη μεγέθους της υπολογιστικής προσπάθειας για αυτούς τους γρήγορους αλγόριθμους είναι μικρότερη από εκατό βήματα για έναν σωλήνα με 20 χάντρες.

## Χώρα προέλευσης θέματος

Ταϊβάν

## 2024-SK-04 – Ακολουθία γραμμάτων

Ηλικιακή ομάδα: Α' - Β' Γυμνασίου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Ακολουθία γραμμάτων

Βάλτε γράμματα σε μια σειρά και κερδίστε πόντους: Αν 2 ίδια γράμματα βρίσκονται ακριβώς το ένα δίπλα στο άλλο, λαμβάνετε 2 πόντους, εάν 3 ίδια γράμματα είναι ακριβώς το ένα δίπλα στο άλλο, παίρνετε 3 πόντους κ.ο.κ.

Παράδειγμα: Για αυτή τη σειρά λαμβάνετε 0 βαθμούς, επειδή δεν υπάρχουν ίδια γράμματα το ένα δίπλα στο άλλο:



Τώρα αντικαταστήστε το πρώτο C με ένα A και το δεύτερο C με ένα B. Παίρνετε 4 βαθμούς για το αποτέλεσμα: 2 για τα δύο A και 2 για τα δύο B στη σειρά.



Τώρα έχετε μια νέα σειρά με 12 γράμματα. Μπορείτε να αντικαταστήσετε οποιαδήποτε τρία γράμματα σε αυτό με τα γράμματα B, B και C.



## Ερώτηση

Αντικαταστήστε τα γράμματα με τέτοιο τρόπο ώστε να λαμβάνετε όσο το δυνατόν περισσότερους πόντους για το αποτέλεσμα. Για να αντικαταστήσετε, σύρετε τα γράμματα B, B και C σε οποιοδήποτε γράμμα της σειράς.

## Απάντηση

A B B C C B B B A A A

## Εξήγηση απάντησης

Παρακάτω είναι η λύση:

A B B C C B B B A A A

Για αυτή τη σειρά παίρνετε  $2 + 2 + 4 + 3 = 11$  πόντους. Από αριστερά προς τα δεξιά υπάρχουν 2 B, 2 C, 4 B και 3 A σε ομάδες το ένα μετά το άλλο. Δεν μπορείτε να κερδίσετε περισσότερους πόντους:

Η σειρά έχει 12 γράμματα, που σημαίνει ότι θα μπορούσατε να κερδίσετε το πολύ 12 πόντους. Ωστόσο, το A στα αριστερά δεν μπορεί να ανήκει σε μια ομάδα, επειδή δεν έχετε A, για να αντικαταστήσετε το B στα δεξιά του. Τα υπόλοιπα 11 γράμματα ανήκουν σε μια ομάδα και το καθένα συνεισφέρει 1 βαθμό στη συνολική βαθμολογία. Ένα γράμμα μπορεί να συνεισφέρει 1 βαθμό.

Υπάρχουν άλλες αλλαγές που μπορούν να σας δώσουν 11 βαθμούς; Για να λάβετε όσο το δυνατόν περισσότερους πόντους, πρέπει να τακτοποιήσετε τα γράμματα B, B και C σε σειρά όπου δημιουργείται μια νέα ομάδα και καμία υπάρχουσα ομάδα δεν διαλύεται ή μειώνεται:

Εάν ένα γράμμα τοποθετηθεί σε μια υπάρχουσα ομάδα, τότε προσθέστε 1 βαθμό. Ωστόσο, εάν ένα γράμμα τοποθετηθεί δίπλα σε ένα μόνο γράμμα για να σχηματιστεί μια νέα ομάδα. Όταν ένα γράμμα τοποθετείται ανάμεσα σε δύο μεμονωμένα γράμματα, έτσι ώστε να σχηματιστεί μια νέα ομάδα των τριών, τότε προστίθενται 3 πόντοι.

Το παράδειγμα δίνει αρχικά 5 βαθμούς. Για να λάβετε 11 βαθμούς, τα B, B και C πρέπει να τοποθετηθούν με τέτοιο τρόπο ώστε να δημιουργείται μια νέα ομάδα των δύο κάθε φορά. Μόνο δύο μονά B και δύο μονά C είναι διαθέσιμα για αυτό:



Υπάρχει μόνο μία δυνατότητα να σχηματιστούν τρεις νέες ομάδες των δύο με τους B, B και C χωρίς να μειωθεί το μέγεθος των υπαρχουσών ομάδων: να κάνουμε. Μόνο για την τελευταία πιθανότητα παίρνεις 11 πόντους, ενώ για τις υπόλοιπες 10 πόντους. Εδώ είναι και οι τέσσερις συνδυασμοί:



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Αυτό το παράδειγμα δείχνει ξεκάθαρα πώς προχωρούν οι επιστήμονες υπολογιστών όταν αναζητούν μια βέλτιστη λύση ή μια μέθοδο για την εξεύρεση λύσης.

Η ιδέα είναι να βρείτε μια τοποθέτηση των γραμμάτων B, B και C που να κερδίζει τους περισσότερους πόντους. Μια πρώτη προσέγγιση θα μπορούσε απλώς να είναι η δημιουργία όλων των δυνατοτήτων στις οποίες τρία γράμματα της σειράς αντικαθίστανται από τα B, B και C. Για μια σειρά δώδεκα γραμμάτων, υπάρχουν  $12 * 11 * 10 = 1320$  δυνατότητες, για καθεμία από τις οποίες η βαθμολογία πρέπει να υπολογιστεί και να συγκριθεί για να προσδιοριστεί η καλύτερη λύση.

Ο αριθμός των λύσεων μπορεί να μειωθεί σημαντικά εάν λάβετε υπόψη ότι τα B, B και C βαθμολογούνται μόνο εάν βρίσκονται δίπλα στα ίδια γράμματα. Εάν λάβετε επίσης υπόψη ότι οι υπάρχουσες ομάδες δεν πρέπει να μειωθούν σε μέγεθος, ο αριθμός των λύσεων μειώνεται περαιτέρω. Εάν λάβετε επίσης υπόψη ότι η σύνδεση μεμονωμένων γραμμάτων φέρνει περισσότερους πόντους από τη μεγέθυνση υπαρχουσών ομάδων, τότε απομένει μόνο μία πιθανή τοποθέτηση.

Οι 1320 δυνατότητες που αναφέρονται παραπάνω δεν είναι πολλές για να υπολογίσει ένας υπολογιστής. Ωστόσο, η επιστήμη των υπολογιστών πάντα ρωτά πώς θα άλλαζε η προσπάθεια που απαιτείται για την επίλυση του προβλήματος καθώς μεγαλώνει το μέγεθος του προβλήματος. Με μεγαλύτερες σειρές και περισσότερα εναλλασσόμενα γράμματα, ο χώρος της λύσης θα ήταν αντίστοιχα μεγαλύτερος. Αξίζει τον κόπο να αναλύσουμε προσεκτικά το πρόβλημα και να αναπτύξουμε μια αποτελεσματική μέθοδο.

Από την άποψη της πληροφορικής, αυτό είναι ένα πρόβλημα βελτιστοποίησης. Στον τομέα της πληροφορικής, η βελτιστοποίηση παίζει καθοριστικό ρόλο. Έχει να κάνει με το να κάνουμε κάτι όσο το δυνατόν πιο αποτελεσματικό ή λειτουργικό εντός ορισμένων περιορισμών. Η βελτιστοποίηση αφορά την εύρεση της καλύτερης από όλες τις πιθανές λύσεις. Για παράδειγμα, όταν μιλάμε για βελτιστοποίηση κώδικα σε έναν μεταγλωττιστή, αναφερόμαστε στη διαδικασία τροποποίησης του κώδικα ώστε να γίνει πιο αποτελεσματικός και μικρότερος χωρίς να αλλάξει η έξοδος του. Αυτό είναι σημαντικό γιατί βοηθά τα προγράμματα να τρέχουν πιο γρήγορα και να χρησιμοποιούν λιγότερους πόρους. Ομοίως, η συμπίεση δεδομένων είναι ένα άλλο παράδειγμα βελτιστοποίησης. Περιλαμβάνει την κωδικοποίηση πληροφοριών χρησιμοποιώντας λιγότερα bit από την αρχική αναπαράσταση για εξοικονόμηση χώρου αποθήκευσης ή μείωση του όγκου των δεδομένων που πρέπει να μεταδοθούν. Σε αυτήν την εργασία, όταν προσπαθούμε να τακτοποιήσουμε βέλτιστα τα γράμματα «BBC», ουσιαστικά κάνουμε το ίδιο πράγμα - προσπαθούμε να βρούμε την πιο αποτελεσματική διάταξη που θα πετύχει τον στόχο μας, δηλαδή να δημιουργήσουμε την ακολουθία που μας δίνει το μεγαλύτερο σκορ. Για να γίνει αυτό, αποθηκεύουμε τα γράμματα του BBC για να αυξήσουμε το μήκος μιας υποακολουθίας των ίδιων γραμμάτων ή να δημιουργήσουμε μια νέα υποακολουθία. Για όλες τις ακολουθίες που δημιουργούμε με αυτόν τον τρόπο, υπολογίζουμε τις βαθμολογίες τους. Στη συνέχεια επιλέγουμε τη σειρά με την καλύτερη βαθμολογία. Όπως και στην πληροφορική, αναζητούμε την «καλύτερη» λύση εντός των περιορισμών που έχουμε θέσει.

### Υπολογιστική σκέψη

Όταν αναζητούμε μια λύση, χρησιμοποιούμε την αποσύνθεση, η οποία περιλαμβάνει τη διάσπαση του προβλήματος σε μικρότερα, πιο διαχειρίσιμα μέρη. Αυτή είναι μια βασική πτυχή της υπολογιστικής σκέψης. Το πρώτο βήμα σε αυτή τη διαδικασία είναι να προσδιορίσουμε όλες τις ακολουθίες που θα μπορούσαν ενδεχομένως να σχετίζονται με τη λύση μας. Αφού έχουμε αυτές τις ακολουθίες, υπολογίζουμε μια βαθμολογία για κάθε μία. Αφού υπολογιστούν όλες οι βαθμολογίες, περνάμε στην αξιολόγηση.

Η αξιολόγηση είναι μια άλλη σημαντική πτυχή της υπολογιστικής σκέψης. Σε αυτήν την περίπτωση, αξιολογούμε τις βαθμολογίες των ακολουθιών μας. Τέλος, επιλέγουμε τη σειρά με την υψηλότερη βαθμολογία. Αυτή είναι η βέλτιστη λύση μας. Αξίζει να σημειωθεί ότι η αξιολόγηση περιλαμβάνει επίσης τον προσδιορισμό των ακολουθιών που αξίζουν να εξεταστούν αρχικά. Χρησιμοποιώντας την αποσύνθεση και την αξιολόγηση εργασιών, μπορούμε συστηματικά και αποτελεσματικά να βρούμε την καλύτερη λύση στο πρόβλημά μας.

## Χώρα προέλευσης θέματος

Σλοβακία

## 2024-CA-02 – Αλυσίδες λέξεων

Ηλικιακή ομάδα: Α' - Β' Γυμνασίου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Αλυσίδες λέξεων

Ο Κάστορας παίζει με κωδικούς που αποτελούνται από τρεις χαρακτήρες. Τοποθετεί τους κωδικούς σε αλυσίδες κωδικών έτσι ώστε μόνο ένας χαρακτήρας να αλλάζει από τον ένα κωδικό στον επόμενο στην αλυσίδα κωδικών.

Για παράδειγμα, οι κωδικοί αυτοί μπορούν να τοποθετηθούν σε μια αλυσίδα κωδικών:  
XUG → XUD → XED → KED

Ο Κάστορας δημιούργησε εννέα κωδικούς: TOF, XEW, TEF, CET, COF, TEW, COT, CEF, και XEF.

Τους τοποθετεί σε αλυσίδες των τριών κωδικών έτσι ώστε κάθε ένας από τους εννέα κωδικούς να χρησιμοποιείται σε μία μόνο αλυσίδα κωδικών.

Καμία από τις παρακάτω αλυσίδες κώδικα δεν παραβιάζει τους κανόνες, αλλά μία από αυτές καθιστά αδύνατη την δημιουργία τριών αλυσίδων κώδικα χωρίς επανάληψη κωδικών.

### Ερώτηση

Ποια επιλογή δεν θα μπορούσε να είναι μία από τις αλυσίδες κώδικα του Κάστορα;

XEW → TEW → TEF

COF → COT → CET

TEF → CEF → COF

CEF → CET → COT

### Απάντηση

TEF → CEF → COF

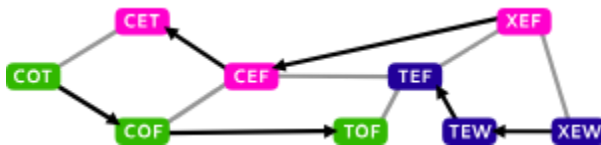
## Εξήγηση απάντησης

Η απάντηση είναι  $TEF \rightarrow CEF \rightarrow COF$ .

Για να λύσετε αυτό το πρόβλημα, είναι χρήσιμο να σχεδιάσετε ένα διάγραμμα. Σε αυτό το διάγραμμα, δύο λέξεις συνδέονται με μια γραμμή εάν μπορούν να διαδέχονται η μία την άλλη σε μια αλυσίδα λέξεων.



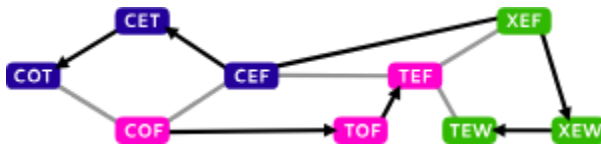
Η επιλογή  $XEW \rightarrow TEW \rightarrow TEF$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $XEF \rightarrow CEF \rightarrow CET$  και  $COT \rightarrow COF \rightarrow TOF$ , όπως φαίνεται.



Η επιλογή  $COF \rightarrow COT \rightarrow CET$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $CEF \rightarrow TEF \rightarrow TOF$  και  $XEF \rightarrow XEW \rightarrow TEW$ , όπως φαίνεται.

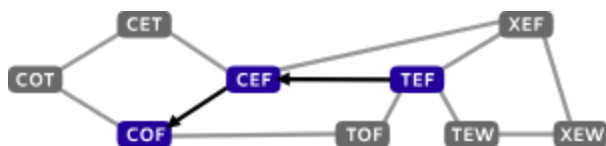


Η επιλογή  $CEF \rightarrow CET \rightarrow COT$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $COF \rightarrow TOF \rightarrow TEF$  και  $XEF \rightarrow XEW \rightarrow TEW$ , όπως φαίνεται.



Η επιλογή  $TEF \rightarrow CEF \rightarrow COF$  δεν είναι μια πιθανή αλυσίδα λέξεων επειδή, για παράδειγμα, στο διάγραμμα, το  $TOF$  συνδέεται μόνο με  $TEF$  και  $COF$ . Επομένως, το  $TOF$  πρέπει να είναι σε μια αλυσίδα λέξεων με τουλάχιστον μία από αυτές τις λέξεις. Δεδομένου ότι το  $TEF$  και το  $COF$  είναι και τα δύο στην αλυσίδα λέξεων για την τρίτη επιλογή, και το  $TOF$  δεν είναι, αυτό δεν αφήνει καμία πιθανή αλυσίδα λέξεων για το  $TOF$ .





## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Στα σχέδια με τα οποία έχουμε εξηγήσει τη λύση, κάθε λέξη αντιστοιχεί σε έναν κόμβο. Μια σύνδεση μεταξύ δύο κόμβων δείχνει ότι αυτές οι δύο λέξεις διαφέρουν ακριβώς σε μία γράμμα. Αυτοί οι δύο κόμβοι είναι τότε γείτονες.

Στην αλυσίδα λέξεων BAD — RAD — RAT — ROT, για παράδειγμα, το BAD και το RAD είναι γείτονες, αλλά το BAD και το RAT δεν είναι. Γενικά ονομάζουμε γράφους, δομές που αποτελούνται από κόμβους και ακμές και τις χρησιμοποιούμε για να χρησιμοποιήσουμε τις ακμές για να αναπαραστήσουμε τις σχέσεις μεταξύ των αντικειμένων που αντιπροσωπεύονται από τους κόμβους.

Η αλυσίδα λέξεων BAD — RAD — RAT περιλαμβάνει τρεις κόμβους και δύο άκρες. Μεταξύ BAD και RAT υπάρχει μια διαδρομή που συνδέει τους αντίστοιχους κόμβους μέσω του κοινού γείτονα RAD. Από καθεμία από τις εννέα λέξεις μπορούμε να φτάσουμε σε όλες τις άλλες λέξεις. Αυτό μπορεί να γίνει απευθείας (δηλ. εάν οι αντίστοιχοι κόμβοι είναι γείτονες) ή χρησιμοποιώντας μια μεγαλύτερη διαδρομή που εκτείνεται σε πολλούς γείτονες. Για να λύσουμε το πρόβλημα του κάστορα χρειαζόμαστε τρία τέτοια μονοπάτια. Κάθε κόμβος πρέπει να χρησιμοποιείται ακριβώς μία φορά.

Τα γραφήματα είναι ένα δημοφιλές εργαλείο που χρησιμοποιείται από επιστήμονες υπολογιστών για τη μοντελοποίηση πολύπλοκων ζητημάτων. Τα προγράμματα μπορούν να λύσουν πολλές εργασίες χρησιμοποιώντας γραφήματα.

Τα συστήματα πλοήγησης, για παράδειγμα, είναι προγράμματα που χρησιμοποιούν πολύ μεγάλα γραφήματα, βήμα προς βήμα από την αφετηρία μέχρι τον προορισμό.

## Χώρα προέλευσης θέματος

Καναδάς

## 2024-IT-01b – Χάρτης φαγητού

Ηλικιακή ομάδα: Α' – Β' Γυμνασίου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Χάρτης φαγητού

Ο Κάστορας Θανάσης έκρυψε τις προμήθειες τροφής του κάτω από 9 από τα 17 δέντρα που περιβάλλουν μέρος της λίμνης. Ο Θανάσης έφτιαξε έναν χάρτη και έγραψε σε κάθε κουτί κατά μήκος της όχθης τον αριθμό των δέντρων που βρίσκονται κοντά του και έχουν κρυμμένη τροφή.

Για παράδειγμα, το κουτί «3» σημαίνει ότι ακριβώς 3 από τα 5 επισημασμένα δέντρα κοντά του έχουν κρυμμένη τροφή.



Αυτό δεν φαίνεται καλή ιδέα στη φίλη του, την Γιώτα και έχει δίκιο!

Στην πραγματικότητα, ο Κάστορας Θανάσης θα μπορέσει να αναγνωρίσει με βεβαιότητα μόνο 7 από τα 9 δέντρα κάτω από τα οποία έκρυψε την τροφή.

### Ερώτηση

Επιλέξτε αυτά τα 7 δέντρα.

Κάντε "Αποθήκευση" όταν τελειώσετε.

### Απάντηση

Η σωστή απάντηση είναι:

	A	B	C	D	E	F	G
1	🌳	★ 🌳	🌳	★ 🌳	🌳	🌳	🌳
2	★ 🌳	2	2	1	1	2	🌳
3	🌳	2				2	★ 🌳
4	★ 🌳	3	2	1		2	🌳
5	🌳	★ 🌳	★ 🌳	1		1	🌳

### Εξήγηση απάντησης

Τα επτά δέντρα κάτω από τα οποία σίγουρα κρύβεται φαγητό είναι αυτά που επισημαίνονται στην παρακάτω εικόνα.

	A	B	C	D	E	F	G
1	🌳	★ 🌳	🌳	★ 🌳	🌳	🌳	🌳
2	★ 🌳	2	2	1	1	2	🌳
3	🌳	2				2	★ 🌳
4	★ 🌳	3	2	1		2	🌳
5	🌳	★ 🌳	★ 🌳	1		1	🌳

Δύο δέντρα με κρυφή τροφή και ένα σίγουρα άδειο (X) μπορούν να αναγνωριστούν ξεκινώντας από το κάτω μέρος.

🌳	🌳	🌳	🌳	🌳	🌳	🌳	🌳
🌳	2	2	1	1	2	🌳	🌳
🌳	2				2	🌳	🌳
🌳	3	2	1		2	🌳	🌳
✕	★ 🌳	★ 🌳	1		1	🌳	🌳

Από κάτω δεξιά προς τα αριστερά: το πρώτο δέντρο πρέπει να έχει τροφή για να ικανοποιήσει το χαμηλότερο 1. Το δεύτερο δέντρο πρέπει επίσης να έχει τροφή για να ικανοποιήσει το χαμηλότερο 2. Αντίθετα, αν υπήρχε τροφή κάτω από το δέντρο στην κάτω αριστερή γωνία, τότε δεν θα ήταν δυνατό να ικανοποιηθεί ο περιορισμός 2 που περικλείεται από τον μικρό κύκλο.

Με παρόμοιο σκεπτικό αποδεικνύεται εύκολα ότι, όσον αφορά το αριστερό μέρος της λιμνούλας, η συνέχεια επιβάλλεται:

		2	2	1
	2			
		3	2	1
				1

Όσον αφορά τη δεξιά πλευρά της λίμνης, υπάρχουν δύο δυνατότητες:

				?
1	1	2		
		2		
		2	?	
		1		

1	1	2	?	
		2		
		2		
		1	?	

Έτσι μόνο ένα ακόμα κόκκινο αστέρι μπορεί να προστεθεί.

Πώς μπορούμε να επισημοποιήσουμε καλύτερα το πρόβλημά μας; Αφού επιλέξουμε τα δύο κουτιά στην τελευταία σειρά που έχουν σίγουρα κρυμμένο φαγητό, δίνουμε ένα όνομα σε κάθε άλλο κουτί που περιέχει ένα δέντρο:

E	F	G	H	I	J	K
D	2	2	1	1	2	L
C	2				2	M
B	3	2	1		2	N
A	★ 🌲	★ 🌲	1		1	O

Οι αριθμοί στα μπλε πλαίσια εκφράζουν περιορισμούς. προχωρώντας δεξιόστροφα από κάτω, οι τρεις πρώτοι (1, 1, 2) είναι ήδη ικανοποιημένοι, για να ικανοποιηθεί ο περιορισμός 3 είναι απαραίτητο να επιλέξετε ακριβώς ένα ακόμη στοιχείο από το σύνολο {A, B, C}, και ούτω καθεξής. Τότε το πρόβλημα μπορεί να επισημοποιηθεί ως εξής:

- από τα {A, B, C} επιλέξτε ακριβώς ένα στοιχείο,
- από τα {B, C, D} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {C, D, E, F, G} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {F, G, H} επιλέξτε ακριβώς 2 στοιχεία,
- από {G, H, I} επιλέξτε ακριβώς ένα στοιχείο,
- από {H, I, J} επιλέξτε ακριβώς ένα στοιχείο,
- από {I, J, K, L, M} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {L, M, N} επιλέξτε ακριβώς 2 στοιχεία,
- από {M, N, O} επιλέξτε ακριβώς 2 στοιχεία,
- από το {N, O} επιλέξτε ακριβώς ένα στοιχείο.

Από εξερεύνηση όλων των πιθανοτήτων (δεν μπορείτε να επιλέξετε το A από το πρώτο σετ, γιατί τότε θα πρέπει να επιλέξετε είτε το B είτε το C από το δεύτερο, αν επιλέξετε B από το πρώτο σετ, τότε πρέπει να επιλέξετε το D από το δεύτερο κ.ο.κ.) μπορεί να επαληθευτεί ότι υπάρχουν δύο λύσεις:

{B, D, F, H, K, M, N}, {B, D, F, H, L, M, O}.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Αυτή η εργασία είναι εμπνευσμένη από ένα διάσημο βιντεοπαιχνίδι παζλ λογικής, γνωστό με διαφορετικά ονόματα (Minesweeper, Flower Field), όπου παίζει και η τύχη. Και ακόμη και όταν ανοίγουν μεγάλες επιφάνειες, μπορεί να προκύψουν καταστάσεις όπου δεν υπάρχουν αρκετές διαθέσιμες πληροφορίες για συνεχίστε το παιχνίδι με ασφάλεια, οπότε στο τέλος πρέπει να βασιστείτε στην τύχη. Αυτή είναι ακριβώς η περίπτωση που προκύπτει στο έργο μας: η δεξιά πλευρά της λίμνης δεν μπορεί να λυθεί με ασφάλεια!

Σημειώστε ότι, γενικά, εάν υπάρχουν  $N$  δέντρα γύρω από τη λίμνη, αντιστοιχούν σε  $2^N$  πιθανές διαμορφώσεις, και τότε είναι συνήθως πολύ πιο εύκολο να ελέγξετε εάν μια δεδομένη λύση είναι συνεπής με τους περιορισμούς του προβλήματος παρά να την βρείτε!

Μέσω αυτής της εργασίας μπορούμε επίσης να δούμε πόσο συχνά ένα πρόβλημα στο σύνολό του μπορεί σταδιακά να αντιμετωπιστεί και να λυθεί σε ξεχωριστά μέρη. Μακράν η πιο ενδιαφέρουσα πτυχή από την άποψη της πληροφορικής είναι η επισημοποίηση του προβλήματος, μέσω μιας κατάλληλης δομής δεδομένων (όπως κάναμε για αυτό το πρόβλημα στο τελευταίο μέρος της εξήγησης), η οποία υποδηλώνει πώς, γενικά, αυτό το πρόβλημα είναι υπολογιστικά δύσκολο, καθώς και η αποδοχή πολλαπλών λύσεων.

### Υπολογιστική Σκέψη

Εάν δεν είστε πολύ άτυχοι, παίζοντας το Flower Field, η λογική παίζει σημαντικό ρόλο και η ερώτηση που τίθεται στο πρόβλημα μας το αποδεικνύει πειστικά: για να φτάσετε στη σωστή απάντηση συλλογίζοντας τους περιορισμούς, είναι απαραίτητο να διερευνήσετε μερικές διαφορετικές δυνατότητες και τις συνέπειες τους. Όμως, ορισμένα προβλήματα μπορεί να μην μπορούν να λυθούν με βεβαιότητα! Αυτός ο τύπος συλλογισμού είναι μεταξύ αυτών που χαρακτηρίζουν την υπολογιστική σκέψη. Ο λογικός συλλογισμός, η δημιουργία υποψήφιων λύσεων και η ικανοποίηση περιορισμών είναι κοινές διαδικασίες στην υπολογιστική σκέψη.

## Χώρα προέλευσης θέματος

Ιταλία

## 2024-SI-01 – Επιστροφή βιβλίων

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Επιστροφή βιβλίων

Επόμενος παρακαλώ! Οι κάστορες στο Holzdorf είναι μανιώδεις αναγνώστες. Για αυτόν τον λόγο υπάρχουν συνήθως πολλοί κάστορες στην ουρά στο γραφείο επιστροφής βιβλίων της βιβλιοθήκης. Ισχύει ο ακόλουθος κανόνας:

Ο κάστορας με τα λιγότερα βιβλία είναι ο επόμενος στην ουρά.

Η βιβλιοθηκάριος χρειάζεται πάντα ακριβώς ένα λεπτό για να επιστρέψει ένα βιβλίο. Μόλις σημειώσει τα βιβλία που επέστρεψε ο ένας κάστορας και τα βάλει στην άκρη, ο επόμενος κάστορας από την ουρά παίρνει τη σειρά του. Ο επόμενος κάστορας είναι πάντα αυτός με τα λιγότερα βιβλία.



Ένα πρωί έρχονται 5 κάστορες και θέλουν να επιστρέψουν τα βιβλία τους. Ο πίνακας δείχνει την ώρα άφιξης και τον αριθμό των βιβλίων για κάθε κάστορα.

Όνομα	Ωρα άφιξης	Αριθμός βιβλίων
Ana	9:00	4
Beti	9:02	6
Cene	9:03	2
Darja	9:05	4
Emil	9:11	1

Η Ana φτάνει πρώτη και μπορεί να επιστρέψει αμέσως τα 4 βιβλία της.

### Ερώτηση

Με ποια σειρά επιστρέφουν οι κάστορες τα βιβλία τους;

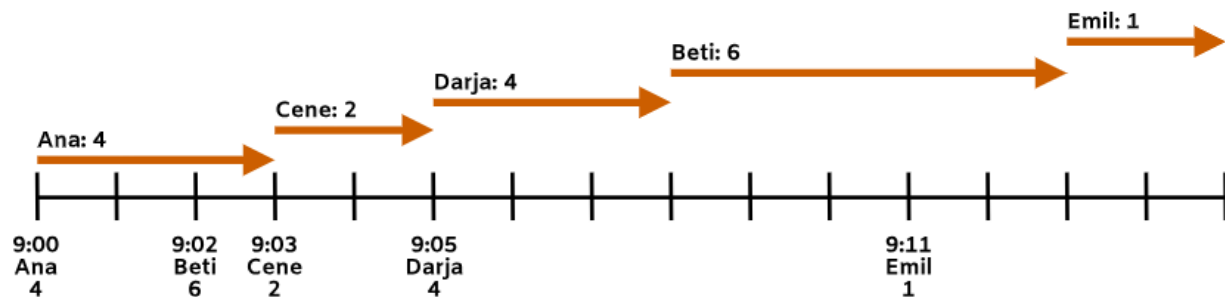
## Απάντηση

6. Ana
7. Cene
8. Darja
9. Beti
10. Emil

## Εξήγηση απάντησης

Η σωστή απάντηση είναι Ana, Cene, Darja, Beti, Emil.

Το παρακάτω διάγραμμα Gantt παρέχει μια οπτική αναπαράσταση του πώς ο βιβλιοθηκάριος αντιμετωπίζει αυτά τα βιβλία που επιστρέφουν και με ποια σειρά.



Αυτή η εικόνα δείχνει πώς επιστρέφονται τα βιβλία:

- Πρώτη η Ana. Χρειάζονται 4 λεπτά για να σημειώσει ο βιβλιοθηκάριος τα 4 βιβλία. Μόνο τότε μπορεί να λάβει τα επόμενα βιβλία.
- Στο μεταξύ, η Beti και η Cene έχουν φτάσει και περιμένουν. Η Cene έχει μόνο 2 βιβλία. Γι' αυτό είναι η επόμενη. Μετά από 2 λεπτά σημειώνονται τα βιβλία της.
- Στο μεταξύ, η Darja έχει έρθει και στέκεται δίπλα στον Beti. Η Darja έχει 4 βιβλία και η Beti έχει 6. Η Darja έχει τα λιγότερα βιβλία και είναι η επόμενη. Η καμμένη η Beti πρέπει ακόμα να περιμένει. Μετά από 4 λεπτά ο βιβλιοθηκάριος έχει τελειώσει με τα βιβλία της Darja και τώρα μπορεί επιτέλους να φροντίσει την Beti. Γιατί είναι η μόνη στην ουρά.
- Στο μεταξύ έφτασε και ο Emil. Είναι ο τελευταίος που πάει.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής



Η εργασία δείχνει πώς λειτουργεί ένας scheduler. Ο scheduler είναι ένα πρόγραμμα που ανήκει στο λειτουργικό σύστημα ενός υπολογιστή. Ενώ ένας υπολογιστής είναι σε λειτουργία, πολλές εργασίες πρέπει να υποβληθούν σε επεξεργασία.

Για κάθε εργασία, ξεκινά μια διαδικασία που λύνει την εργασία. Ο Scheduler ελέγχει τη ροή των διεργασιών. Καθορίζει πότε και για πόσο χρόνο εκτελείται μια διεργασία από την κεντρική μονάδα επεξεργασίας του υπολογιστή (CPU)

Υπάρχουν διαφορετικές στρατηγικές που χρησιμοποιούν οι scheduler:

- Priority Scheduling – Σε αυτόν τον τύπο ουράς, κάθε διεργασία έχει μια προτεραιότητα, και η CPU εκτελεί πρώτα τις διεργασίες με την υψηλότερη προτεραιότητα. Αν δύο διεργασίες έχουν την ίδια προτεραιότητα, χρησιμοποιείται μια δευτερεύουσα στρατηγική για την επιλογή.
- First Come First Serve (FCFS) – Οι διεργασίες εκτελούνται με τη σειρά που φτάνουν στην ουρά, χωρίς προτεραιότητες ή διακοπές. Αυτό σημαίνει ότι η πρώτη διεργασία που φτάνει στη CPU εκτελείται μέχρι να ολοκληρωθεί, προτού ξεκινήσει η επόμενη.
- Round Robin – Η CPU εκτελεί κάθε διεργασία για ένα συγκεκριμένο χρονικό διάστημα (χρονικό τεμάχιο - time quantum) και στη συνέχεια την τοποθετεί στο τέλος της ουράς αν δεν έχει ολοκληρωθεί, εξασφαλίζοντας δίκαιη κατανομή του χρόνου επεξεργασίας.
- Shortest Job First (SJF) – Η CPU εκτελεί πρώτα τη διεργασία με τον μικρότερο χρόνο εκτέλεσης, μειώνοντας τον μέσο χρόνο αναμονής. Αν δύο διεργασίες έχουν τον ίδιο χρόνο εκτέλεσης, επιλέγεται η πρώτη που έφτασε στην ουρά.

Υπολογιστική σκέψη

Σε αυτήν την εργασία, μας δίνεται ένα σύνολο κανόνων για το πώς λειτουργεί η βιβλιοθήκη, συμπεριλαμβανομένου του χρόνου που χρειάζεται ο βιβλιοθηκάριος για να επεξεργαστεί μια επιστροφή βιβλίου, πότε κάποιος μπορεί να φύγει από την ουρά και ποιος θα εξυπηρετείται στη συνέχεια. Αυτό το σύνολο κανόνων είναι στην πραγματικότητα ένας αλγόριθμος και ακολουθώντας προσεκτικά αυτόν τον αλγόριθμο στα δεδομένα που παρέχονται (η λίστα των δανειοληπτών βιβλίων και τα στοιχεία τους) μπορούμε να λύσουμε το πρόβλημα. Ο αλγόριθμος έχει πιθανές ασάφειες, όπως τι πρέπει να γίνει στην κατάσταση που δύο δανειολήπτες βιβλίων στην ουρά έχουν ταυτόχρονα τον ίδιο αριθμό βιβλίων, αλλά αυτή η κατάσταση δεν χρειάζεται να οριστεί για αυτό το συγκεκριμένο σενάριο.

## Χώρα προέλευσης θέματος

Σλοβενία

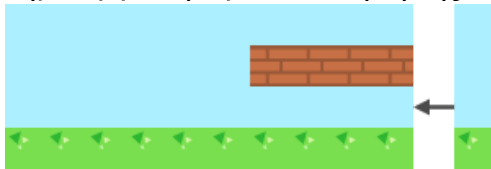
## 2024-DE-07 – Ηλεκτρονικό παιχνίδι

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

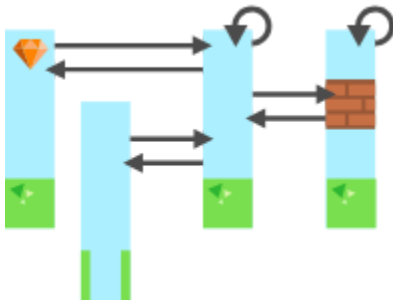
Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Ηλεκτρονικό παιχνίδι

Σε ένα ηλεκτρονικό παιχνίδι, το φόντο αποτελείται από μια ακολουθία πλακιδίων. Ο υπολογιστής προσθέτει συνεχώς ένα νέο πλακίδιο στα δεξιά της ακολουθίας και ταυτόχρονα αφαιρεί ένα πλακίδιο από τα αριστερά. Με αυτόν τον τρόπο, ο υπολογιστής δημιουργεί την ψευδαίσθηση της κίνησης.



Ο υπολογιστής επιλέγει ένα νέο πλακίδιο προς προσθήκη χρησιμοποιώντας το διάγραμμα στα δεξιά. Κοιτάζει το προηγούμενο πλακίδιο και ελέγχει τα βέλη που προέρχονται από αυτό. Στη συνέχεια επιλέγει τυχαία ένα από τα πλακίδια στο οποίο δείχνει ένα από τα βέλη.



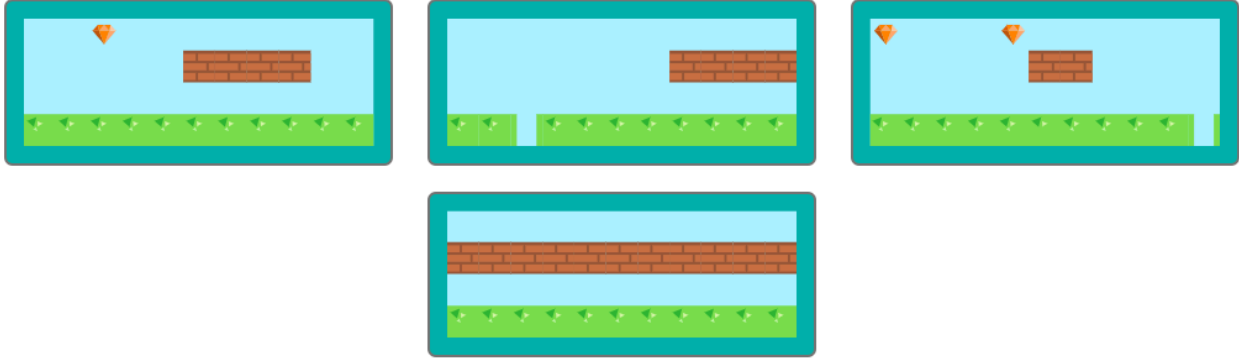
Για παράδειγμα, μετά το  πλακίδιο, ο υπολογιστής μπορεί να διαλέξει είτε το  ή το



πλακίδιο.

## Ερώτηση

Μία από τις παρακάτω εικόνες ΔΕΝ αποτελεί έγκυρο φόντο σε αυτό το παιχνίδι. Ποια από αυτές είναι;

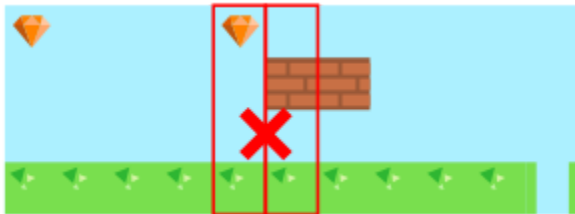



## Απάντηση

Απάντηση 3

## Εξήγηση απάντησης

Το φόντο που φαίνεται στην ακόλουθη απάντηση δεν έχει δημιουργηθεί σύμφωνα με το διάγραμμα.



Ένα πλακίδιο με διαμάντι μπορεί να ακολουθήσει μόνο το  πλακίδιο, κάτι που δεν συμβαίνει εδώ.

Υπάρχουν διάφοροι τρόποι για να βρείτε τη λύση σε αυτό το ερώτημα. Μια απλή στρατηγική είναι να ελέγξετε οποιαδήποτε δεδομένη εικόνα φόντου και να συγκρίνετε κάθε πλακίδιο με το διάγραμμα για να δείτε αν είναι έγκυρο. Ο πιο γρήγορος τρόπος είναι αυτός:

- Ελέγξτε τα πλακίδια στο διάγραμμα, ξεκινώντας από τα πλακίδια που έχουν μόνο ένα βέλος. Αυτά τα βέλη μπορούν να θεωρηθούν ως περιορισμοί.  $A \rightarrow B$  σημαίνει «το πλακίδιο A πρέπει να ακολουθείται από το πλακίδιο B».

- Στη συνέχεια ελέγξτε εάν αυτός ο περιορισμός ισχύει για κάθε εικόνα φόντου.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Ορισμένα παιχνίδια στον υπολογιστή – όπως τα ατελείωτα παιχνίδια δρομέων – έχουν φόντο που κινείται οριζόντια, δημιουργώντας την ψευδαίσθηση ότι ο παίκτης κινείται σε έναν κόσμο φαντασίας. Μερικές φορές το φόντο δεν είναι μια σταθερή εικόνα, αλλά δημιουργείται αυτόματα από τον υπολογιστή. Αυτό ονομάζεται procedural generation. Συνήθως, μικρότερα στοιχεία συνδυάζονται τυχαία για να δημιουργήσουν μια μεγάλη ποικιλία από φόντο.

Ωστόσο, ο συνδυασμός στοιχείων δεν μπορεί να είναι εντελώς τυχαίος, αλλά πρέπει να ακολουθεί ορισμένους κανόνες.

Στην ερώτηση, αυτοί οι κανόνες αντιπροσωπεύονται από ένα διάγραμμα πλακιδίων και τέτοια διαγράμματα, όπου τα στοιχεία συνδέονται με βέλη, ονομάζονται κατευθυνόμενα γραφήματα. Τα στοιχεία των διαγραμμάτων ονομάζονται κόμβοι και τα βέλη είναι κατευθυνόμενες άκρες. Τα κατευθυνόμενα γραφήματα χρησιμοποιούνται για όλους τους τύπους μοντελοποίησης.

Τα πεπερασμένα αυτόματα πηγαίνουν ένα βήμα παραπέρα: Έχουν καταστάσεις, κατάσταση έναρξης, καταστάσεις λήξης και κανόνες για το πώς να μετακινηθείτε από τη μια κατάσταση στην άλλη. Τα πεπερασμένα αυτόματα χρησιμοποιούνται συχνά για μοντελοποίηση στην επιστήμη των υπολογιστών.

### Υπολογιστική σκέψη

Για να λύσετε την άσκηση πρέπει να διαβάσετε και να χρησιμοποιήσετε το κατευθυνόμενο γράφημα. Το γράφημα καθορίζει τη σύνθεση της εικόνας φόντου. Μπορεί να θεωρηθεί ως μια αφαίρεση του μηχανισμού που δημιουργεί αυτόματα εικόνες.

Επιπλέον, η αναγνώριση μοτίβων μπορεί να χρησιμοποιηθεί για να χωρίσει οπτικά το φόντο σε μια σειρά πλακιδίων και να ανιχνεύσει ελαττωματικές μεταβάσεις. Επομένως, αυτή η άσκηση περιλαμβάνει αλγοριθμική σκέψη, αφαίρεση και αναγνώριση προτύπων.

## Χώρα προέλευσης θέματος

Γερμανία

## 2024-HU-03 – Κρυμμένος θησαυρός

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Κρυμμένος θησαυρός

Ο Γιώργος ο Πειρατής ψάχνει για κρυμμένο θησαυρό σε ένα νησί.

Ο Γιώργος έχει έναν χάρτη που δείχνει πού βρίσκεται ο θησαυρός. Ο χάρτης χωρίζεται σε 16 τετράγωνα. Ο Γιώργος μπορεί να εισάγει οποιονδήποτε αριθμό περιοχών σε μια ειδική συσκευή και η συσκευή θα τον ενημερώσει εάν ο θησαυρός βρίσκεται σε μία από τις περιοχές ή όχι.

Για παράδειγμα, εάν η συσκευή εμφανίζει "Ναι" όταν ο Γιώργος εισέρχεται στις παρακάτω επισημασμένες περιοχές, τότε ο θησαυρός βρίσκεται σε οποιαδήποτε από αυτές τις τρεις περιοχές.



Ο Γιώργος θέλει να μάθει όσο πιο γρήγορα γίνεται σε ποιο τετράγωνο βρίσκεται ο θησαυρός.

### Ερώτηση

Ποιος είναι ο ελάχιστος αριθμός φορών που θα πρέπει ο Γιώργος να χρησιμοποιήσει τη συσκευή για να βρει που είναι ο θησαυρός;

### Απάντηση

## Εξήγηση απάντησης

Η σωστή απάντηση είναι 4.

Ο Γιώργος πρέπει να εισάγει πρώτα τον μισό χάρτη στη συσκευή. Υπάρχουν δύο σενάρια.

A. Εάν η συσκευή υποδεικνύει ότι ο θησαυρός βρίσκεται στις δεδομένες περιοχές, ο Γιώργος τώρα διαιρεί περαιτέρω αυτήν την περιοχή σε δύο μισά και εισάγει το ένα μισό στη συσκευή.

B. Εάν η συσκευή υποδεικνύει ότι ο θησαυρός δεν βρίσκεται στις δεδομένες περιοχές, χωρίζει τις υπόλοιπες περιοχές σε δύο μισά και εισάγει ένα από τα μισά στη συσκευή.

Ο Γιώργος συνεχίζει τα παραπάνω μέχρι να βρει την περιοχή που έχει τον θησαυρό.

Για παράδειγμα, εάν εισάγει A,B,C,D,E,F,G,H στη συσκευή και επιστρέψει ένα "όχι", θα μπορούσε να εισαγάγει K,L,O,P στη συσκευή (οποιοσδήποτε υπόλοιπες 4 περιοχές). Εάν επιστρέψει ξανά "όχι", διαιρεί περαιτέρω τις υπόλοιπες περιοχές και εισάγει οποιοσδήποτε από τις υπόλοιπες περιοχές στη συσκευή (για παράδειγμα J,N όπως φαίνεται). Εάν η συσκευή επιστρέψει "όχι", ο θησαυρός είναι κρυμμένος σε μία από τις υπόλοιπες περιοχές I ή M. Εάν εισαγάγει το M και η συσκευή επιστρέψει όχι, τώρα γνωρίζει με βεβαιότητα ότι ο θησαυρός βρίσκεται στην περιοχή I.

Για να γνωρίζει με βεβαιότητα σε ποιο τετράγωνο βρίσκεται ο θησαυρός μετά από τέσσερις ερωτήσεις, ο Γιώργος μπορεί να χρησιμοποιήσει αυτήν τη στρατηγική: Ψάχνει ακριβώς τα μισά τετράγωνα στα οποία ο θησαυρός θα μπορούσε να βρίσκεται, σύμφωνα με τις προηγούμενες απαντήσεις της μηχανής.

A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D
E	F	G	H	E	F	G	H	E	F	G	H	E	F	G	H
I	J	K	L	I	J	K	L	I	J	K	L	I	J	K	L
M	N	O	P	M	N	O	P	M	N	O	P	M	N	O	P

Ο Γιώργος δεν μπορεί να τα βγάλει πέρα με λιγότερες ερωτήσεις. Αν δεν χώριζε τα υπόλοιπα τετράγωνα σε δύο ίσα μισά, ο θησαυρός θα μπορούσε να βρίσκεται στο μεγαλύτερο μέρος. Για να ρωτήσει για αυτό το μέρος, ο Γιώργος θα έπρεπε να κάνει τουλάχιστον τόσες ερωτήσεις όσες και για το μισό.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Η μέθοδος που χρησιμοποιεί ο Γιώργος για το κυνήγι του θησαυρού ονομάζεται δυαδική αναζήτηση (binary search) στην επιστήμη των υπολογιστών. Ο όρος binary προέρχεται

από τη λατινική λέξη bis (δύο φορές). Στη δυαδική αναζήτηση ενός αντικείμενου σε ένα σύνολο, το σύνολο μειώνεται στο μισό ξανά και ξανά από μια σειρά ερωτήσεων, δηλαδή χωρίζεται σε δύο μέρη - εξ ου και "δυαδικό". Στη συνέχεια, υπάρχει ένα μεσαίο αντικείμενο στο σετ και μπορείτε να συγκρίνετε το μεσαίο αντικείμενο με αυτό που ψάχνετε. Εάν το μεσαίο αντικείμενο δεν είναι ήδη αυτό που ψάχνετε, ξέρετε τουλάχιστον σε ποιο μισό σύνολο βρίσκεται το αντικείμενο που αναζητάτε και θα αναζητήστε ξανά αυτό το μισό ακολουθώντας το δυαδικό σύστημα. Με αυτόν τον τρόπο μπορείτε πολύ γρήγορα να φτάσετε στο αντικείμενο που αναζητάτε. Για 1.000 αντικείμενα, χρειάζονται περίπου 10 βήματα αναζήτησης, για 1.000.000 αντικείμενα, περίπου 20. Γενικά, μπορούμε να πούμε ότι για  $n$  αντικείμενα, χρειάζονται περίπου  $\log_2(n)$  βήματα. Η συνάρτηση καταγραφής είναι ο "λογάριθμος των δύο" ή ο λογάριθμος στη βάση 2. Επειδή η δυαδική αναζήτηση είναι τόσο γρήγορη, χρησιμοποιείται συχνά σε προγράμματα υπολογιστών για αναζήτηση σε ταξινομημένα δεδομένα.

Σε αυτήν την άσκηση, ο χώρος αναζήτησης είναι το σύνολο των τετραγώνων στην κάτοψη του νησιού. Τα τετράγωνα μπορούν να τακτοποιηθούν αριθμώντας τα από πάνω προς τα κάτω και από αριστερά προς τα δεξιά. Εάν τα τετράγωνα δεν αριθμηθούν θα είναι λίγο πιο δύσκολο να θυμηθεί κανείς ποιο σύνολο τετραγώνων είναι ακόμα δυνατό για το επόμενο βήμα αναζήτησης και ποιο πρέπει να μειωθεί ξανά στο μισό.

## Χώρα προέλευσης θέματος

Ουγγαρία

# 2024-TW-03-UK – Το πιο μακρύ βραχιόλι

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Το πιο μακρύ βραχιόλι

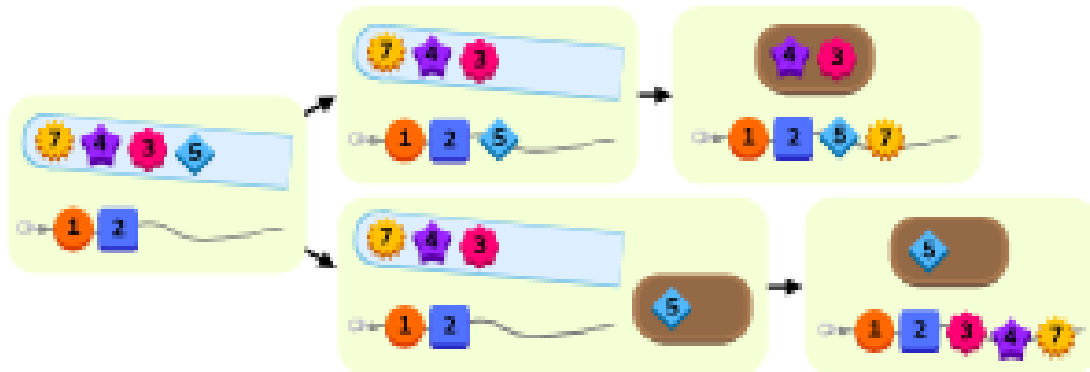
Ο Μάριος φτιάχνει ένα βραχιόλι. Παίρνει χάντρες με αριθμούς από ένα γυάλινο σωλήνα. Κολλάει μερικές χάντρες από αυτές και βάζει τις άλλες στο καφέ κουτί και δεν τις χρησιμοποιεί. Επιτρέπεται να περάσει μια χάντρα μόνο αν:

το κορδόνι είναι άδειο, ή

η χάντρα έχει μεγαλύτερο αριθμό από την τελευταία χάντρα στο κορδόνι.

Σε αυτό το παράδειγμα, η τελευταία χάντρα στο κορδόνι είναι η χάντρα 2. Στη συνέχεια, ο Μάριος μπορεί να περάσει τη χάντρα 5 από τον σωλήνα. Μπορεί επίσης να την αφήσει στο καφέ κουτί και να μην τη χρησιμοποιήσει.

Αν περάσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με τέσσερις χάντρες 1257. Αν δεν χρησιμοποιήσει τη χάντρα 5, μπορεί να φτιάξει ένα βραχιόλι με περισσότερες χάντρες 12347.



Ο Μάριος φτιάχνει ένα καινούργιο βραχιόλι από τις χάντρες αυτού του σωλήνα:





## Ερώτηση

Ποιος είναι ο μέγιστος αριθμός χαντρών που μπορεί να περάσει από το κορδόνι;

## Απάντηση

6

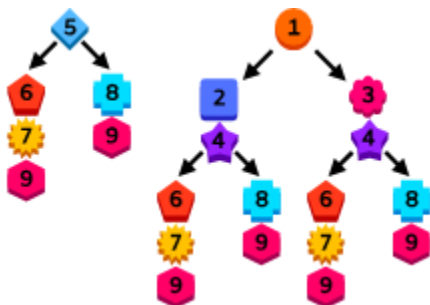
## Εξήγηση απάντησης

Η απάντηση είναι 6.

Θα μπορούσαμε να απαριθμήσουμε όλα τα βραχιόλια για αυτό το σωληνάριο και να δούμε ποια έχουν τον μεγαλύτερο αριθμό χαντρών. Αλλά αυτό θα ήταν χρονοβόρο. Ας ρίξουμε μια πιο προσεκτική ματιά στους αριθμούς:



Η πρώτη χάντρα είναι ο αριθμός 5. Όταν αυτή περνάει στο κορδόνι, οι επόμενες δυνατές χάντρες είναι μόνο οι 6, 7, 8 και 9, επομένως οι πιθανές ακολουθίες θα ήταν 5679 και 589. Εάν η χάντρα 5 αφηθεί στην άκρη και η επόμενη χάντρα, η χάντρα 1, μπει στο κορδόνι, υπάρχουν περισσότερες πιθανές ακολουθίες. Αυτό μπορεί να αναπαρασταθεί ξεκάθαρα σε ένα δέντρο αποφάσεων:



Εάν η χάντρα 1 αφηθεί επίσης στην άκρη και μία διαφορετική χάντρα περαστεί πρώτα, οι πιθανές αλληλουχίες περιέχονται ήδη σε μία από τις παραπάνω ακολουθίες. Εάν ξεκινήσετε με την χάντρα 2, για παράδειγμα, θα λάβετε 24679 ή 2489, που περιέχονται στις μεγαλύτερες ακολουθίες 124679 και 12489 αντίστοιχα.

Τα βραχιόλια με τον μεγαλύτερο αριθμό χαντρών ξεκινούν με τη χάντρα 1 και αποτελούνται από 6 χάντρες: 124679 ή 123679.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Κάθε σωλήνας περιέχει μια ακολουθία αριθμημένων χαντρών. Εάν ο Μάριος ακολουθήσει τον δεύτερο κανόνα (μόνο οι χάντρες στο κορδόνι που έχουν μεγαλύτερο αριθμό από την τελευταία), θα πάρει ένα βραχιόλι με αύξουσα ακολουθία. Για να κατασκευαστεί ένα βραχιόλι με όσο το δυνατόν περισσότερες χάντρες, πρέπει να καθορίσει η μεγαλύτερη ανοδική ακολουθία.

Για μεγάλες ακολουθίες, θα χρειαζόταν πολύς χρόνος για να σχηματιστούν όλες οι πιθανές αύξουσες υποακολουθίες και στη συνέχεια να προσδιοριστεί η μεγαλύτερη. Εάν ο σωλήνας περιείχε 20 χάντρες, για παράδειγμα, η υπολογιστική προσπάθεια θα ήταν της τάξης ενός εκατομμυρίου βημάτων.

Ευτυχώς, στην επιστήμη των υπολογιστών έχουν αναπτυχθεί αλγόριθμοι που μπορούν να βρουν τη μεγαλύτερη αύξουσα υποακολουθία πιο γρήγορα. Αυτό γίνεται χρησιμοποιώντας μια τεχνική που ονομάζεται δυναμικός προγραμματισμός. Η τάξη μεγέθους της υπολογιστικής προσπάθειας για αυτούς τους γρήγορους αλγόριθμους είναι μικρότερη από εκατό βήματα για έναν σωλήνα με 20 χάντρες.

## Χώρα προέλευσης θέματος

Ταϊβάν

## 2024-SK-04 – Ακολουθία γραμμάτων

Ηλικιακή ομάδα: Γ' Γυμνασίου - Α' Λυκείου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Ακολουθία γραμμάτων

Βάλτε γράμματα σε μια σειρά και κερδίστε πόντους: Αν 2 ίδια γράμματα βρίσκονται ακριβώς το ένα δίπλα στο άλλο, λαμβάνετε 2 πόντους, εάν 3 ίδια γράμματα είναι ακριβώς το ένα δίπλα στο άλλο, παίρνετε 3 πόντους κ.ο.κ.

Παράδειγμα: Για αυτή τη σειρά λαμβάνετε 0 βαθμούς, επειδή δεν υπάρχουν ίδια γράμματα το ένα δίπλα στο άλλο:



Τώρα αντικαταστήστε το πρώτο C με ένα A και το δεύτερο C με ένα B. Παίρνετε 4 βαθμούς για το αποτέλεσμα: 2 για τα δύο A και 2 για τα δύο B στη σειρά.



Τώρα έχετε μια νέα σειρά με 12 γράμματα. Μπορείτε να αντικαταστήσετε οποιαδήποτε τρία γράμματα σε αυτό με τα γράμματα B, B και C.



## Ερώτηση

Αντικαταστήστε τα γράμματα με τέτοιο τρόπο ώστε να λαμβάνετε όσο το δυνατόν περισσότερους πόντους για το αποτέλεσμα. Για να αντικαταστήσετε, σύρετε τα γράμματα B, B και C σε οποιοδήποτε γράμμα της σειράς.

## Απάντηση



## Εξήγηση απάντησης

Παρακάτω είναι η λύση:



Για αυτή τη σειρά παίρνετε  $2 + 2 + 4 + 3 = 11$  πόντους. Από αριστερά προς τα δεξιά υπάρχουν 2 B, 2 C, 4 B και 3 A σε ομάδες το ένα μετά το άλλο. Δεν μπορείτε να κερδίσετε περισσότερους πόντους:

Η σειρά έχει 12 γράμματα, που σημαίνει ότι θα μπορούσατε να κερδίσετε το πολύ 12 πόντους. Ωστόσο, το A στα αριστερά δεν μπορεί να ανήκει σε μια ομάδα, επειδή δεν έχετε A, για να αντικαταστήσετε το B στα δεξιά του. Τα υπόλοιπα 11 γράμματα ανήκουν σε μια ομάδα και το καθένα συνεισφέρει 1 βαθμό στη συνολική βαθμολογία. Ένα γράμμα μπορεί να συνεισφέρει 1 βαθμό.

Υπάρχουν άλλες αλλαγές που μπορούν να σας δώσουν 11 βαθμούς; Για να λάβετε όσο το δυνατόν περισσότερους πόντους, πρέπει να τακτοποιήσετε τα γράμματα B, B και C σε σειρά όπου δημιουργείται μια νέα ομάδα και καμία υπάρχουσα ομάδα δεν διαλύεται ή μειώνεται:

Εάν ένα γράμμα τοποθετηθεί σε μια υπάρχουσα ομάδα, τότε προσθέστε 1 βαθμό. Ωστόσο, εάν ένα γράμμα τοποθετηθεί δίπλα σε ένα μόνο γράμμα για να σχηματιστεί μια νέα ομάδα. Όταν ένα γράμμα τοποθετείται ανάμεσα σε δύο μεμονωμένα γράμματα, έτσι ώστε να σχηματιστεί μια νέα ομάδα των τριών, τότε προστίθενται 3 πόντοι.

Το παράδειγμα δίνει αρχικά 5 βαθμούς. Για να λάβετε 11 βαθμούς, τα B, B και C πρέπει να τοποθετηθούν με τέτοιο τρόπο ώστε να δημιουργείται μια νέα ομάδα των δύο κάθε φορά. Μόνο δύο μονά B και δύο μονά C είναι διαθέσιμα για αυτό:



Υπάρχει μόνο μία δυνατότητα να σχηματιστούν τρεις νέες ομάδες των δύο με τους B, B και C χωρίς να μειωθεί το μέγεθος των υπαρχουσών ομάδων: να κάνουμε. Μόνο για την τελευταία πιθανότητα παίρνεις 11 πόντους, ενώ για τις υπόλοιπες 10 πόντους. Εδώ είναι και οι τέσσερις συνδυασμοί:



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Αυτό το παράδειγμα δείχνει ξεκάθαρα πώς προχωρούν οι επιστήμονες υπολογιστών όταν αναζητούν μια βέλτιστη λύση ή μια μέθοδο για την εξεύρεση λύσης.

Η ιδέα είναι να βρείτε μια τοποθέτηση των γραμμάτων B, B και C που να κερδίζει τους περισσότερους πόντους. Μια πρώτη προσέγγιση θα μπορούσε απλώς να είναι η δημιουργία όλων των δυνατοτήτων στις οποίες τρία γράμματα της σειράς αντικαθίστανται από τα B, B και C. Για μια σειρά δώδεκα γραμμάτων, υπάρχουν  $12 * 11 * 10 = 1320$  δυνατότητες, για καθεμία από τις οποίες η βαθμολογία πρέπει να υπολογιστεί και να συγκριθεί για να προσδιοριστεί η καλύτερη λύση.

Ο αριθμός των λύσεων μπορεί να μειωθεί σημαντικά εάν λάβετε υπόψη ότι τα B, B και C βαθμολογούνται μόνο εάν βρίσκονται δίπλα στα ίδια γράμματα. Εάν λάβετε επίσης υπόψη ότι οι υπάρχουσες ομάδες δεν πρέπει να μειωθούν σε μέγεθος, ο αριθμός των λύσεων μειώνεται περαιτέρω. Εάν λάβετε επίσης υπόψη ότι η σύνδεση μεμονωμένων γραμμάτων φέρνει περισσότερους πόντους από τη μεγέθυνση υπαρχουσών ομάδων, τότε απομένει μόνο μία πιθανή τοποθέτηση.

Οι 1320 δυνατότητες που αναφέρονται παραπάνω δεν είναι πολλές για να υπολογίσει ένας υπολογιστής. Ωστόσο, η επιστήμη των υπολογιστών πάντα ρωτά πώς θα άλλαζε η προσπάθεια που απαιτείται για την επίλυση του προβλήματος καθώς μεγαλώνει το μέγεθος του προβλήματος. Με μεγαλύτερες σειρές και περισσότερα εναλλασσόμενα γράμματα, ο χώρος της λύσης θα ήταν αντίστοιχα μεγαλύτερος. Αξίζει τον κόπο να αναλύσουμε προσεκτικά το πρόβλημα και να αναπτύξουμε μια αποτελεσματική μέθοδο.

Από την άποψη της πληροφορικής, αυτό είναι ένα πρόβλημα βελτιστοποίησης. Στον τομέα της πληροφορικής, η βελτιστοποίηση παίζει καθοριστικό ρόλο. Έχει να κάνει με το να κάνουμε κάτι όσο το δυνατόν πιο αποτελεσματικό ή λειτουργικό εντός ορισμένων περιορισμών. Η βελτιστοποίηση αφορά την εύρεση της καλύτερης από όλες τις πιθανές λύσεις. Για παράδειγμα, όταν μιλάμε για βελτιστοποίηση κώδικα σε έναν μεταγλωττιστή, αναφερόμαστε στη διαδικασία τροποποίησης του κώδικα ώστε να γίνει πιο αποτελεσματικός και μικρότερος χωρίς να αλλάξει η έξοδος του. Αυτό είναι σημαντικό γιατί βοηθά τα προγράμματα να τρέχουν πιο γρήγορα και να χρησιμοποιούν λιγότερους πόρους. Ομοίως, η συμπίεση δεδομένων είναι ένα άλλο παράδειγμα βελτιστοποίησης. Περιλαμβάνει την κωδικοποίηση πληροφοριών χρησιμοποιώντας λιγότερα bit από την αρχική αναπαράσταση για εξοικονόμηση χώρου αποθήκευσης ή μείωση του όγκου των δεδομένων που πρέπει να μεταδοθούν. Σε αυτήν την εργασία, όταν προσπαθούμε να τακτοποιήσουμε βέλτιστα τα γράμματα «BBC», ουσιαστικά κάνουμε το ίδιο πράγμα - προσπαθούμε να βρούμε την πιο αποτελεσματική διάταξη που θα πετύχει τον στόχο μας, δηλαδή να δημιουργήσουμε την ακολουθία που μας δίνει το μεγαλύτερο σκορ. Για να γίνει αυτό, αποθηκεύουμε τα γράμματα του BBC για να αυξήσουμε το μήκος μιας υποακολουθίας των ίδιων γραμμάτων ή να δημιουργήσουμε μια νέα υποακολουθία. Για όλες τις ακολουθίες που δημιουργούμε με αυτόν τον τρόπο, υπολογίζουμε τις βαθμολογίες τους. Στη συνέχεια επιλέγουμε τη σειρά με την καλύτερη βαθμολογία. Όπως και στην πληροφορική, αναζητούμε την «καλύτερη» λύση εντός των περιορισμών που έχουμε θέσει.

### Υπολογιστική σκέψη

Όταν αναζητούμε μια λύση, χρησιμοποιούμε την αποσύνθεση, η οποία περιλαμβάνει τη διάσπαση του προβλήματος σε μικρότερα, πιο διαχειρίσιμα μέρη. Αυτή είναι μια βασική πτυχή της υπολογιστικής σκέψης. Το πρώτο βήμα σε αυτή τη διαδικασία είναι να προσδιορίσουμε όλες τις ακολουθίες που θα μπορούσαν ενδεχομένως να σχετίζονται με τη λύση μας. Αφού έχουμε αυτές τις ακολουθίες, υπολογίζουμε μια βαθμολογία για κάθε μία. Αφού υπολογιστούν όλες οι βαθμολογίες, περνάμε στην αξιολόγηση.

Η αξιολόγηση είναι μια άλλη σημαντική πτυχή της υπολογιστικής σκέψης. Σε αυτήν την περίπτωση, αξιολογούμε τις βαθμολογίες των ακολουθιών μας. Τέλος, επιλέγουμε τη σειρά με την υψηλότερη βαθμολογία. Αυτή είναι η βέλτιστη λύση μας. Αξίζει να σημειωθεί ότι η αξιολόγηση περιλαμβάνει επίσης τον προσδιορισμό των ακολουθιών που αξίζουν να εξεταστούν αρχικά. Χρησιμοποιώντας την αποσύνθεση και την αξιολόγηση εργασιών, μπορούμε συστηματικά και αποτελεσματικά να βρούμε την καλύτερη λύση στο πρόβλημά μας.

## Χώρα προέλευσης θέματος

Σλοβακία

## 2024-CA-02 – Αλυσίδες λέξεων

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Αλυσίδες λέξεων

Ο Κάστορας παίζει με κωδικούς που αποτελούνται από τρεις χαρακτήρες. Τοποθετεί τους κωδικούς σε αλυσίδες κωδικών έτσι ώστε μόνο ένας χαρακτήρας να αλλάζει από τον ένα κωδικό στον επόμενο στην αλυσίδα κωδικών.

Για παράδειγμα, οι κωδικοί αυτοί μπορούν να τοποθετηθούν σε μια αλυσίδα κωδικών:  
XUG → XUD → XED → KED

Ο Κάστορας δημιούργησε εννέα κωδικούς: TOF, XEW, TEF, CET, COF, TEW, COT, CEF, και XEF.

Τους τοποθετεί σε αλυσίδες των τριών κωδικών έτσι ώστε κάθε ένας από τους εννέα κωδικούς να χρησιμοποιείται σε μία μόνο αλυσίδα κωδικών.

Καμία από τις παρακάτω αλυσίδες κώδικα δεν παραβιάζει τους κανόνες, αλλά μία από αυτές καθιστά αδύνατη την δημιουργία τριών αλυσίδων κώδικα χωρίς επανάληψη κωδικών.

### Ερώτηση

Ποια επιλογή δεν θα μπορούσε να είναι μία από τις αλυσίδες κώδικα του Κάστορα;

XEW → TEW → TEF

COF → COT → CET

TEF → CEF → COF

CEF → CET → COT

### Απάντηση

TEF → CEF → COF



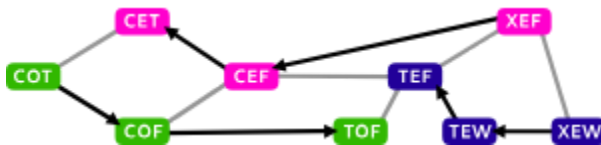
## Εξήγηση απάντησης

Η απάντηση είναι  $TEF \rightarrow CEF \rightarrow COF$ .

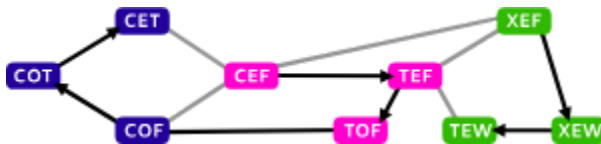
Για να λύσετε αυτό το πρόβλημα, είναι χρήσιμο να σχεδιάσετε ένα διάγραμμα. Σε αυτό το διάγραμμα, δύο λέξεις συνδέονται με μια γραμμή εάν μπορούν να διαδέχονται η μία την άλλη σε μια αλυσίδα λέξεων.



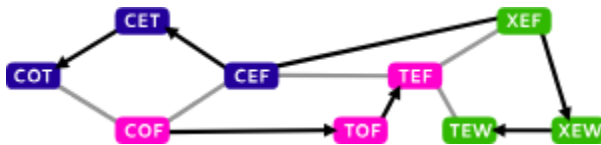
Η επιλογή  $XEW \rightarrow TEW \rightarrow TEF$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $XEF \rightarrow CEF \rightarrow CET$  και  $COT \rightarrow COF \rightarrow TOF$ , όπως φαίνεται.



Η επιλογή  $COF \rightarrow COT \rightarrow CET$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $CEF \rightarrow TEF \rightarrow TOF$  και  $XEF \rightarrow XEW \rightarrow TEW$ , όπως φαίνεται.



Η επιλογή  $CEF \rightarrow CET \rightarrow COT$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $COF \rightarrow TOF \rightarrow TEF$  και  $XEF \rightarrow XEW \rightarrow TEW$ , όπως φαίνεται.



Η επιλογή  $TEF \rightarrow CEF \rightarrow COF$  δεν είναι μια πιθανή αλυσίδα λέξεων επειδή, για παράδειγμα, στο διάγραμμα, το  $TOF$  συνδέεται μόνο με  $TEF$  και  $COF$ . Επομένως, το  $TOF$  πρέπει να είναι σε μια αλυσίδα λέξεων με τουλάχιστον μία από αυτές τις λέξεις. Δεδομένου ότι το  $TEF$  και το  $COF$  είναι και τα δύο στην αλυσίδα λέξεων για την τρίτη επιλογή, και το  $TOF$  δεν είναι, αυτό δεν αφήνει καμία πιθανή αλυσίδα λέξεων για το  $TOF$ .



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Στα σχέδια με τα οποία έχουμε εξηγήσει τη λύση, κάθε λέξη αντιστοιχεί σε έναν κόμβο. Μια σύνδεση μεταξύ δύο κόμβων δείχνει ότι αυτές οι δύο λέξεις διαφέρουν ακριβώς σε μία γράμμα. Αυτοί οι δύο κόμβοι είναι τότε γείτονες.

Στην αλυσίδα λέξεων BAD — RAD — RAT — ROT, για παράδειγμα, το BAD και το RAD είναι γείτονες, αλλά το BAD και το RAT δεν είναι. Γενικά ονομάζουμε γράφους, δομές που αποτελούνται από κόμβους και ακμές και τις χρησιμοποιούμε για να χρησιμοποιήσουμε τις ακμές για να αναπαραστήσουμε τις σχέσεις μεταξύ των αντικειμένων που αντιπροσωπεύονται από τους κόμβους.

Η αλυσίδα λέξεων BAD — RAD — RAT περιλαμβάνει τρεις κόμβους και δύο άκρες. Μεταξύ BAD και RAT υπάρχει μια διαδρομή που συνδέει τους αντίστοιχους κόμβους μέσω του κοινού γείτονα RAD. Από καθεμία από τις εννέα λέξεις μπορούμε να φτάσουμε σε όλες τις άλλες λέξεις. Αυτό μπορεί να γίνει απευθείας (δηλ. εάν οι αντίστοιχοι κόμβοι είναι γείτονες) ή χρησιμοποιώντας μια μεγαλύτερη διαδρομή που εκτείνεται σε πολλούς γείτονες. Για να λύσουμε το πρόβλημα του κάστορα χρειαζόμαστε τρία τέτοια μονοπάτια. Κάθε κόμβος πρέπει να χρησιμοποιείται ακριβώς μία φορά.

Τα γραφήματα είναι ένα δημοφιλές εργαλείο που χρησιμοποιείται από επιστήμονες υπολογιστών για τη μοντελοποίηση πολύπλοκων ζητημάτων. Τα προγράμματα μπορούν να λύσουν πολλές εργασίες χρησιμοποιώντας γραφήματα.

Τα συστήματα πλοήγησης, για παράδειγμα, είναι προγράμματα που χρησιμοποιούν πολύ μεγάλα γραφήματα, βήμα προς βήμα από την αφετηρία μέχρι τον προορισμό.

## Χώρα προέλευσης θέματος

Καναδάς

## 2024-LT-05 – Φίλοι

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Φίλοι

Υπάρχουν επτά κάστορες σε μια τάξη με τα ονόματα: Alex, Bess, Cora, Dave, Eric, Fred, και Gigi. Μερικοί από τους κάστορες γνωρίζονται μεταξύ τους και άλλοι όχι.

Το παρακάτω διάγραμμα δείχνει ποιοι κάστορες γνωρίζονται μεταξύ τους με ένα πράσινο σημάδι.



	Alex	Bess	Cora	Dave	Eric	Fred	Gigi
Alex	■	✓	✓				
Bess	✓	■					✓
Cora	✓		■		✓		
Dave				■		✓	
Eric			✓		■		
Fred				✓		■	
Gigi		✓					■

Κάθε κάστορας έλαβε ένα διαφορετικό μήνυμα από κάποιον εκτός της τάξης και το μοιράστηκε με όλους τους κάστορες που γνωρίζει στην τάξη του. Κάθε φορά που ένας κάστορας λαμβάνει ένα νέο μήνυμα από κάποιον στην τάξη, το μοιράζεται αμέσως με όλους τους υπόλοιπους κάστορες που γνωρίζει στην τάξη.

### Ερώτηση

Ποιος ή ποιοι κάστορες στην τάξη έλαβαν τον λιγότερο αριθμό μηνυμάτων συνολικά;

### Απάντηση

Dave, Fred

## Εξήγηση απάντησης

Η σωστή απάντηση είναι ο Dave και ο Fred.

Μπορούμε να χρησιμοποιήσουμε ένα διάγραμμα για να δείξουμε τις σχέσεις μεταξύ των καστόρων στην τάξη. Σε αυτό το διάγραμμα, κάθε κάστορας αντιπροσωπεύεται από έναν κύκλο, και αν οι δύο κάστορες γνωρίζονται μεταξύ τους, τότε οι κύκλοι τους συνδέονται με μια γραμμή.

Ο Dave γνωρίζει μόνο τον Fred και ο Fred μόνο τον Dave, επομένως συνδέουμε τους κύκλους τους μεταξύ τους και δεν τους συνδέουμε με άλλους κύκλους. Ο Eric γνωρίζει μόνο την Cora, οπότε συνδέουμε τους κύκλους τους. Η Cora γνωρίζει επίσης τον Alex, οπότε συνδέουμε τους κύκλους τους. Ο Άλεξ γνωρίζει επίσης την Bess, οπότε συνδέουμε τους κύκλους τους. Τέλος, η Bess γνωρίζει και την Gigi, οπότε συνδέουμε τους κύκλους τους.



Τώρα από το διάγραμμα, σαφώς καθένας από τους Dave και Fred μπορεί να λάβει μόνο 2 μηνύματα, ενώ ο Eric, η Cora, ο Alex, η Bess και η Gigi θα λάβει 5 μηνύματα.

Επομένως, ο Dave και ο Fred θα λάβουν τον λιγότερο αριθμό μηνυμάτων.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Οι γράφοι είναι ένα σημαντικό εργαλείο αφαίρεσης στην επιστήμη των υπολογιστών. Σε αυτήν την εργασία, ένα γράφημα παρουσιάζεται ως πίνακας γειννίας. Το πρόβλημα των φιλιών με κάστορες σχετίζεται με ένα πρόβλημα συνδεσιμότητας γράφου. Στη θεωρία γράφων και στην επιστήμη των υπολογιστών, ένας πίνακας γειννίας είναι ένας τετραγωνικός πίνακας που χρησιμοποιείται για να αναπαραστήσει ένα πεπερασμένο γράφο. Τα στοιχεία του πίνακα υποδεικνύουν εάν ζεύγη στοιχείων είναι γειτονικά ή όχι στο γράφημα. Ο γράφος σε αυτήν την εργασία είναι μη κατευθυνόμενος (δηλαδή όλες οι ακμές του είναι αμφίδρομες), επειδή ο πίνακας γειννίας είναι συμμετρικός.

Ο πίνακας γειτνίασης μπορεί να χρησιμοποιηθεί ως δομή δεδομένων για την αναπαράσταση γράφων σε προγράμματα υπολογιστών για το χειρισμό γραφημάτων. Η κύρια εναλλακτική δομή δεδομένων, που χρησιμοποιείται επίσης για αυτήν την εφαρμογή, είναι η λίστα γειτνίασης.

## Χώρα προέλευσης θέματος

Λιθουανία

## 2024-PK-03 – Σιδηροδρομικό δίκτυο

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Σιδηροδρομικό δίκτυο

Στη χώρα της Bebrania, οι γειτονικοί οικισμοί συνδέονται με ένα δίκτυο σιδηροδρομικών γραμμών. Για κάθε γραμμή, υπάρχει ένας περιορισμένος αριθμός τρένων που μπορούν να ταξιδεύουν κατά μήκος της κάθε μέρα, όπως φαίνεται στο παρακάτω διάγραμμα. Διαφορετικά τρένα μπορεί να κινούνται στην ίδια γραμμή.



Η πόλη Q προσφέρθηκε να στείλει υλικά στην πόλη P. Προς αυτή την κατεύθυνση, τα τρένα πρέπει πάντα να ακολουθούν τα βέλη.

### Ερώτηση

Ποιος είναι ο μέγιστος αριθμός τρένων που μπορούν να αναχωρούν από την πόλη Q και να φτάνουν στην πόλη P κάθε μέρα;

13

15

19

22

### Απάντηση

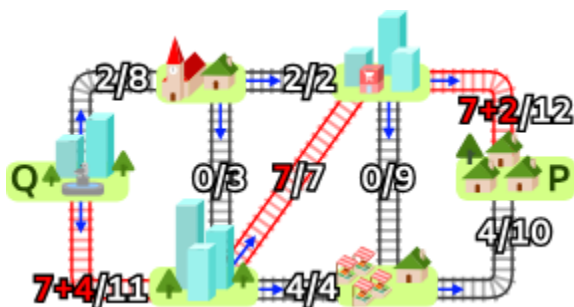
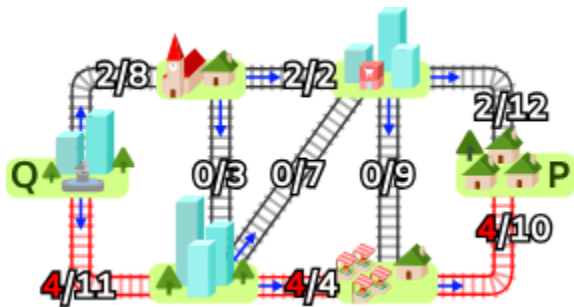
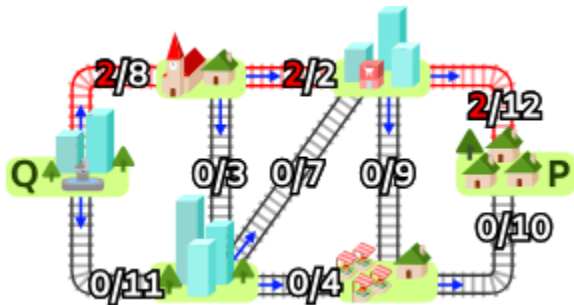
13

## Εξήγηση απάντησης

Η σωστή απάντηση είναι 13.

Η πρώτη παρατήρηση είναι ότι η πόλη Q δεν μπορεί να στείλει στην πόλη P περισσότερα από  $8+11=19$  τρένα, ο αριθμός των τρένων που μπορούν να φύγουν από την πόλη Q. Επομένως, η απάντηση "22" είναι λανθασμένη.

Ένας τρόπος για να λυθεί αυτό το πρόβλημα είναι να βρείτε μια συντομότερη δυνατή διαδρομή από την πόλη Q στην πόλη P (στην περίπτωση αυτή, μήκους 3) και να καθορίσετε τον μεγαλύτερο αριθμό τρένων που μπορεί να φιλοξενήσει. Στη συνέχεια επαναλαμβάνουμε αυτό το βήμα στο υπόλοιπο δίκτυο με τον υπόλοιπο αριθμό τρένων που είναι διαθέσιμα σε κάθε γραμμή. Τα παρακάτω διαγράμματα δείχνουν τρία μονοπάτια που βρέθηκαν με αυτόν τον τρόπο, με συνδυασμένη χωρητικότητα  $2+4+7=13$  τρένα. Υπάρχουν άλλοι τρόποι για να στείλετε 13 τρένα από την πόλη Q στην πόλη P, συμπεριλαμβανομένων ορισμένων που χρησιμοποιούν μεγαλύτερα μονοπάτια.



Για να ελέγξετε ότι το 13 είναι το μέγιστο, παρατηρήστε τα τρία ίχνη στη μέση του δικτύου με χωρητικότητες 2, 7 και 4: όλα τα μονοπάτια περνούν από ένα από αυτά, επομένως δεν μπορούμε να κάνουμε καλύτερα από  $2+7+4=13$ .

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Το πρόβλημα σε αυτήν την εργασία είναι ένα παράδειγμα του προβλήματος μέγιστης ροής. Προβλήματα αυτού του τύπου είναι σαν ένα παιχνίδι παζλ με το νερό να ρέει μέσα από σωλήνες: έχετε ένα δίκτυο σωλήνων με διαφορετικές χωρητικότητες και θέλετε να υπολογίσετε πόσο νερό μπορεί να ρέει από μια πηγή σε μία έξοδο.

Το πρόβλημα μέγιστης ροής μπορεί να λυθεί χρησιμοποιώντας διάφορους αλγόριθμους: Αλγόριθμος Ford-Fulkerson, Αλγόριθμος Edmonds-Karp, Αλγόριθμος Dinic και πολλοί άλλοι. Τα αναπόφευκτα κομμάτια στη μέση του δικτύου σε αυτήν την εργασία σχηματίζουν αυτό που είναι γνωστό ως «ελάχιστη περικοπή».

Καθένας από τους παραπάνω αλγόριθμους έχει τα πλεονεκτήματα και τα μειονεκτήματά του όσον αφορά την απόδοση χρόνου εκτέλεσης, την ευκολία εφαρμογής του και την εφαρμογή σε διαφορετικούς τύπους δικτύων. Η επιλογή του αλγορίθμου εξαρτάται από παράγοντες όπως το μέγεθος του δικτύου, η φύση των χωρητικότητων και το επιθυμητό επίπεδο βελτιστοποίησης.

Οι αλγόριθμοι μέγιστης ροής χρησιμοποιούνται στη βελτιστοποίηση της κυκλοφορίας, στα συστήματα διανομής (π.χ. νερό και ηλεκτρική ενέργεια), στην κατανομή πόρων, στη δικτύωση υπολογιστών και πολλά άλλα.

### Υπολογιστική σκέψη

Η επίλυση αυτής της εργασίας περιλαμβάνει αλγοριθμική σκέψη και αξιολόγηση. Η λύση του προβλήματος αποσυντίθεται σε μια ακολουθία βημάτων, καθένα από τα οποία αποτελείται από την εύρεση μιας διαδρομής μεταξύ Q και P που έχει μέγιστη χωρητικότητα. Ο αλγόριθμος τερματίζεται όταν δεν υπάρχει άλλη διαθέσιμη τέτοια διαδρομή.

## Χώρα προέλευσης θέματος

Πακιστάν



## 2024-DE-06a – Μηχανή μπαλονιών

Ηλικιακή ομάδα: Γ' Γυμνασίου – Α' Λυκείου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

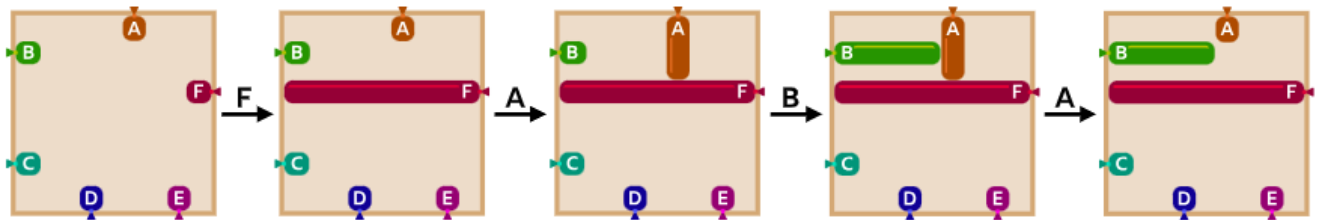
Θέμα : Μηχανή Μπαλονιών

Το μηχάνημα διαβάζει γράμματα ένα-ένα. Όταν διαβάζει ένα γράμμα:

Εάν το μπαλόνι που φέρει την ετικέτα με αυτό το γράμμα είναι ξεφουσκωτό, φουσκώνει μέχρι να αγγίξει ένα άλλο μπαλόνι ή την απέναντι άκρη του πλαισίου.

Διαφορετικά ξεφουσκώνει το μπαλόνι που φέρει την ετικέτα με αυτό το γράμμα.

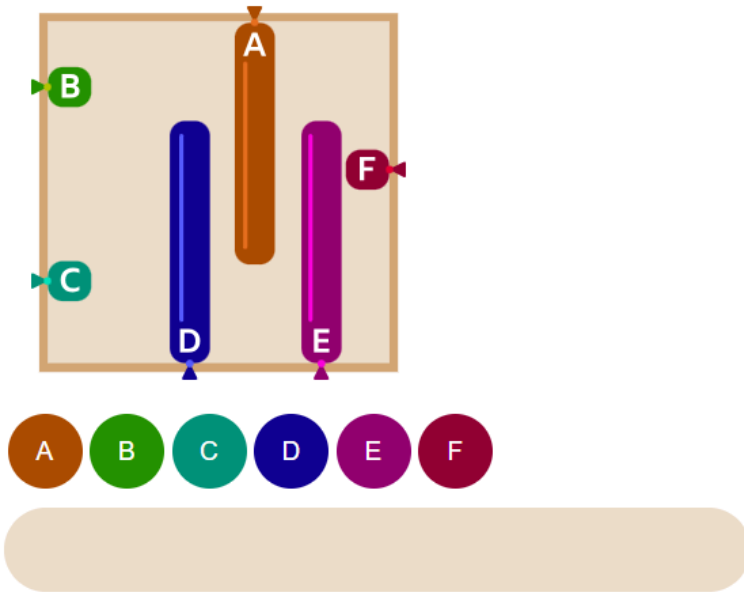
Για παράδειγμα, εάν όλα τα μπαλόνια είναι ξεφουσκωτά στην αρχή και το μηχάνημα γράφει F, A, B και μετά A, θα κάνει τα εξής:



### Ερώτηση

Για να ξεκινήσετε, κάθε μπαλόνι ξεφουσκώνεται και στη συνέχεια το μηχάνημα διαβάζει μια ακολουθία εννέα γραμμάτων. Το αποτέλεσμα είναι η παρακάτω εικόνα. Εισαγάγετε την ακολουθία των εννέα γραμμάτων.

Σύρετε τα γράμματα στη γραμμή για να σχηματίσετε την ακολουθία.



### Απάντηση

BEBCACBDB

BECBACBDB

BEBCABCDB

BECBABCDB

### Εξήγηση απάντησης

Υπάρχουν 4 πιθανές σωστές απαντήσεις με εννέα γράμματα:

BEBCACBDB

BECBACBDB

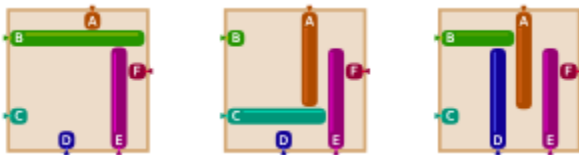
BEBCABCDB

BECBABCDB

Δείχνουμε λεπτομερή εξήγηση για μία απάντηση: B E B C A C B D B. Οι παρακάτω εικόνες απεικονίζουν την εκτέλεση αυτών των οδηγιών. Δείχνουν ενδιάμεσες καταστάσεις. Η πρώτη εικόνα δείχνει την κατάσταση μετά την εκτέλεση του B E. Μπορείτε να δείτε ότι είναι απαραίτητο να διογκωθεί το B πριν από το E για να γίνει το B κατάλληλο φράγμα για το φούσκωμα του E.

Η δεύτερη εικόνα δείχνει την κατάσταση μετά το B E B C A. Το μπαλόνι C χρησιμεύει ως φράγμα για το A και επομένως πρέπει να φουσκώσει πριν από το A. Επιπλέον το μπαλόνι B πρέπει να ξεφουσκώσει, πριν μπορέσει να φουσκώσει το A, διαφορετικά δεν θα φτάσει στο απαιτούμενο μήκος.

Η τρίτη εικόνα δείχνει την κατάσταση μετά το B E B C A C B D. Το B χρησιμεύει ως φράγμα για το D και το C πρέπει να ξεφουσκώσει πριν φουσκώσει το D.



Για να βρείτε όλες τις σωστές απαντήσεις, είναι χρήσιμο να αναπαραστήσετε το πρόβλημα χρησιμοποιώντας ένα κατευθυνόμενο γράφημα. Τα γράμματα αντιπροσωπεύουν τα μπαλόνια. Ένα βέλος από το B στο E υποδεικνύει ότι το μπαλόνι B πρέπει να φουσκώσει πριν από το μπαλόνι E κ.λπ. Υπάρχουν δύο περιπτώσεις όπου η σειρά των B και C μπορεί να ποικίλλει (υποδεικνύεται με διακεκομμένο ορθογώνιο).

Κάθε σωστή απάντηση (ακολουθία μπαλονιών που θα φουσκώσουν) είναι μια τοπολογική σειρά αυτού του γραφήματος. Σε αυτή την περίπτωση έχουμε 4 τοπολογικές ταξινομήσεις οπότε υπάρχουν τέσσερις πιθανές σωστές λύσεις.



Παρατήρηση για την αξιολόγηση της ορθότητας της λύσης σας: τα φουσκωμένα μπαλόνια πρέπει να εμφανίζονται περιττοί χρόνοι → A, D και E πρέπει να εμφανίζονται περιττοί χρόνοι. Τα ξεφουσκωμένα μπαλόνια πρέπει να εμφανίζονται ζυγές φορές (συμπεριλαμβανομένου του 0) → B, C και F.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Η ακολουθία των γραμμάτων είναι ένα πρόγραμμα υπολογιστή, το οποίο ελέγχει μια μηχανή. Κάθε γράμμα είναι μια δήλωση που κάνει το μηχάνημα να φουσκώνει ή να ξεφουσκώνει ένα μπαλόνι. Όπως στα περισσότερα προγράμματα υπολογιστών αυτού του τύπου, η σειρά των δηλώσεων είναι απαραίτητη. Για παράδειγμα, η ακολουθία B E κάνει το μηχάνημα να δημιουργήσει μια διαφορετική εικόνα από την ακολουθία E B.

Ο κατευθυνόμενος γράφος που χρησιμοποιείται στην εξήγηση της απάντησης δεν έχει κύκλους και επομένως ονομάζεται κατευθυνόμενο άκυκλο γράφημα (DAG). Οποιοδήποτε DAG έχει τουλάχιστον μία τοπολογική διάταξη και είναι γνωστοί γρήγοροι αλγόριθμοι για την κατασκευή μιας τοπολογικής ταξινόμησης. Τα DAG έχουν πολυάριθμες εφαρμογές στην πληροφορική, π.χ. προγραμματισμός εργασιών.

## Χώρα προέλευσης θέματος

Γερμανία

## 2024-IT-01b – Χάρτης φαγητού

Ηλικιακή ομάδα: Γ' Γυμνασίου - Α' Λυκείου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Χάρτης φαγητού

Ο Κάστορας Θανάσης έκρυψε τις προμήθειες τροφής του κάτω από 9 από τα 17 δέντρα που περιβάλλουν μέρος της λίμνης. Ο Θανάσης έφτιαξε έναν χάρτη και έγραψε σε κάθε κουτί κατά μήκος της όχθης τον αριθμό των δέντρων που βρίσκονται κοντά του και έχουν κρυμμένη τροφή.

Για παράδειγμα, το κουτί «3» σημαίνει ότι ακριβώς 3 από τα 5 επισημασμένα δέντρα κοντά του έχουν κρυμμένη τροφή.



Αυτό δεν φαίνεται καλή ιδέα στη φίλη του, την Γιώτα και έχει δίκιο! Στην πραγματικότητα, ο Κάστορας Θανάσης θα μπορούσε να αναγνωρίσει με βεβαιότητα μόνο 7 από τα 9 δέντρα κάτω από τα οποία έκρυψε την τροφή.

### Ερώτηση

Επιλέξτε αυτά τα 7 δέντρα.  
Κάντε "Αποθήκευση" όταν τελειώσετε.

### Απάντηση

Η σωστή απάντηση είναι:



### Εξήγηση απάντησης

Τα επτά δέντρα κάτω από τα οποία σίγουρα κρύβεται φαγητό είναι αυτά που επισημαίνονται στην παρακάτω εικόνα.



Δύο δέντρα με κρυφή τροφή και ένα σίγουρα άδειο (X) μπορούν να αναγνωριστούν ξεκινώντας από το κάτω μέρος.



Από κάτω δεξιά προς τα αριστερά: το πρώτο δέντρο πρέπει να έχει τροφή για να ικανοποιήσει το χαμηλότερο 1. Το δεύτερο δέντρο πρέπει επίσης να έχει τροφή για να ικανοποιήσει το χαμηλότερο 2. Αντίθετα, αν υπήρχε τροφή κάτω από το δέντρο στην κάτω αριστερή γωνία, τότε δεν θα ήταν δυνατό να ικανοποιηθεί ο περιορισμός 2 που περικλείεται από τον μικρό κύκλο.

Με παρόμοιο σκεπτικό αποδεικνύεται εύκολα ότι, όσον αφορά το αριστερό μέρος της λιμνούλας, η συνέχεια επιβάλλεται:

		2	2	1
	2			
		3	2	1
				1

Όσον αφορά τη δεξιά πλευρά της λίμνης, υπάρχουν δύο δυνατότητες:

				?
1	1	2		
		2		
		2	?	
		1		

1	1	2	?	
		2		
		2		
		1	?	

Έτσι μόνο ένα ακόμα κόκκινο αστέρι μπορεί να προστεθεί.

Πώς μπορούμε να επισημοποιήσουμε καλύτερα το πρόβλημά μας; Αφού επιλέξουμε τα δύο κουτιά στην τελευταία σειρά που έχουν σίγουρα κρυμμένο φαγητό, δίνουμε ένα όνομα σε κάθε άλλο κουτί που περιέχει ένα δέντρο:

E	F	G	H	I	J	K
D	2	2	1	1	2	L
C	2				2	M
B	3	2	1		2	N
A	★ 🌲	★ 🌲	1		1	O

Οι αριθμοί στα μπλε πλαίσια εκφράζουν περιορισμούς. προχωρώντας δεξιόστροφα από κάτω, οι τρεις πρώτοι (1, 1, 2) είναι ήδη ικανοποιημένοι, για να ικανοποιηθεί ο περιορισμός 3 είναι απαραίτητο να επιλέξετε ακριβώς ένα ακόμη στοιχείο από το σύνολο {A, B, C}, και ούτω καθεξής. Τότε το πρόβλημα μπορεί να επισημοποιηθεί ως εξής:

- από τα {A, B, C} επιλέξτε ακριβώς ένα στοιχείο,
- από τα {B, C, D} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {C, D, E, F, G} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {F, G, H} επιλέξτε ακριβώς 2 στοιχεία,
- από {G, H, I} επιλέξτε ακριβώς ένα στοιχείο,
- από {H, I, J} επιλέξτε ακριβώς ένα στοιχείο,
- από {I, J, K, L, M} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {L, M, N} επιλέξτε ακριβώς 2 στοιχεία,
- από {M, N, O} επιλέξτε ακριβώς 2 στοιχεία,
- από το {N, O} επιλέξτε ακριβώς ένα στοιχείο.

Από εξερεύνηση όλων των πιθανοτήτων (δεν μπορείτε να επιλέξετε το A από το πρώτο σετ, γιατί τότε θα πρέπει να επιλέξετε είτε το B είτε το C από το δεύτερο, αν επιλέξετε B από το πρώτο σετ, τότε πρέπει να επιλέξετε το D από το δεύτερο κ.ο.κ.) μπορεί να επαληθευτεί ότι υπάρχουν δύο λύσεις:

{B, D, F, H, K, M, N}, {B, D, F, H, L, M, O}.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής



Αυτή η εργασία είναι εμπνευσμένη από ένα διάσημο βιντεοπαιχνίδι παζλ λογικής, γνωστό με διαφορετικά ονόματα (Minesweeper, Flower Field), όπου παίζει και η τύχη. Και ακόμη και όταν ανοίγουν μεγάλες επιφάνειες, μπορεί να προκύψουν καταστάσεις όπου δεν υπάρχουν αρκετές διαθέσιμες πληροφορίες για συνεχίστε το παιχνίδι με ασφάλεια, οπότε στο τέλος πρέπει να βασιστείτε στην τύχη. Αυτή είναι ακριβώς η περίπτωση που προκύπτει στο έργο μας: η δεξιά πλευρά της λίμνης δεν μπορεί να λυθεί με ασφάλεια!

Σημειώστε ότι, γενικά, εάν υπάρχουν  $N$  δέντρα γύρω από τη λίμνη, αντιστοιχούν σε  $2^N$  πιθανές διαμορφώσεις, και τότε είναι συνήθως πολύ πιο εύκολο να ελέγξετε εάν μια δεδομένη λύση είναι συνεπής με τους περιορισμούς του προβλήματος παρά να την βρείτε!

Μέσω αυτής της εργασίας μπορούμε επίσης να δούμε πόσο συχνά ένα πρόβλημα στο σύνολό του μπορεί σταδιακά να αντιμετωπιστεί και να λυθεί σε ξεχωριστά μέρη. Μακράν η πιο ενδιαφέρουσα πτυχή από την άποψη της πληροφορικής είναι η επισημοποίηση του προβλήματος, μέσω μιας κατάλληλης δομής δεδομένων (όπως κάναμε για αυτό το πρόβλημα στο τελευταίο μέρος της εξήγησης), η οποία υποδηλώνει πώς, γενικά, αυτό το πρόβλημα είναι υπολογιστικά δύσκολο, καθώς και η αποδοχή πολλαπλών λύσεων.

### Υπολογιστική Σκέψη

Εάν δεν είστε πολύ άτυχοι, παίζοντας το Flower Field, η λογική παίζει σημαντικό ρόλο και η ερώτηση που τίθεται στο πρόβλημα μας το αποδεικνύει πειστικά: για να φτάσετε στη σωστή απάντηση συλλογίζοντας τους περιορισμούς, είναι απαραίτητο να διερευνήσετε μερικές διαφορετικές δυνατότητες και τις συνέπειες τους. Όμως, ορισμένα προβλήματα μπορεί να μην μπορούν να λυθούν με βεβαιότητα! Αυτός ο τύπος συλλογισμού είναι μεταξύ αυτών που χαρακτηρίζουν την υπολογιστική σκέψη. Ο λογικός συλλογισμός, η δημιουργία υποψήφιων λύσεων και η ικανοποίηση περιορισμών είναι κοινές διαδικασίες στην υπολογιστική σκέψη.

## Χώρα προέλευσης θέματος

Ιταλία

## 2024-SK-04 – Ακολουθία γραμμάτων

Ηλικιακή ομάδα: Β' - Γ' Λυκείου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Ακολουθία γραμμάτων

Βάλτε γράμματα σε μια σειρά και κερδίστε πόντους: Αν 2 ίδια γράμματα βρίσκονται ακριβώς το ένα δίπλα στο άλλο, λαμβάνετε 2 πόντους, εάν 3 ίδια γράμματα είναι ακριβώς το ένα δίπλα στο άλλο, παίρνετε 3 πόντους κ.ο.κ.

Παράδειγμα: Για αυτή τη σειρά λαμβάνετε 0 βαθμούς, επειδή δεν υπάρχουν ίδια γράμματα το ένα δίπλα στο άλλο:



Τώρα αντικαταστήστε το πρώτο C με ένα A και το δεύτερο C με ένα B. Παίρνετε 4 βαθμούς για το αποτέλεσμα: 2 για τα δύο A και 2 για τα δύο B στη σειρά.



Τώρα έχετε μια νέα σειρά με 12 γράμματα. Μπορείτε να αντικαταστήσετε οποιαδήποτε τρία γράμματα σε αυτό με τα γράμματα B, B και C.



## Ερώτηση

Αντικαταστήστε τα γράμματα με τέτοιο τρόπο ώστε να λαμβάνετε όσο το δυνατόν περισσότερους πόντους για το αποτέλεσμα. Για να αντικαταστήσετε, σύρετε τα γράμματα B, B και C σε οποιοδήποτε γράμμα της σειράς.

## Απάντηση



## Εξήγηση απάντησης

Παρακάτω είναι η λύση:



Για αυτή τη σειρά παίρνετε  $2 + 2 + 4 + 3 = 11$  πόντους. Από αριστερά προς τα δεξιά υπάρχουν 2 B, 2 C, 4 B και 3 A σε ομάδες το ένα μετά το άλλο. Δεν μπορείτε να κερδίσετε περισσότερους πόντους:

Η σειρά έχει 12 γράμματα, που σημαίνει ότι θα μπορούσατε να κερδίσετε το πολύ 12 πόντους. Ωστόσο, το A στα αριστερά δεν μπορεί να ανήκει σε μια ομάδα, επειδή δεν έχετε A, για να αντικαταστήσετε το B στα δεξιά του. Τα υπόλοιπα 11 γράμματα ανήκουν σε μια ομάδα και το καθένα συνεισφέρει 1 βαθμό στη συνολική βαθμολογία. Ένα γράμμα μπορεί να συνεισφέρει 1 βαθμό.

Υπάρχουν άλλες αλλαγές που μπορούν να σας δώσουν 11 βαθμούς; Για να λάβετε όσο το δυνατόν περισσότερους πόντους, πρέπει να τακτοποιήσετε τα γράμματα B, B και C σε σειρά όπου δημιουργείται μια νέα ομάδα και καμία υπάρχουσα ομάδα δεν διαλύεται ή μειώνεται:

Εάν ένα γράμμα τοποθετηθεί σε μια υπάρχουσα ομάδα, τότε προσθέστε 1 βαθμό. Ωστόσο, εάν ένα γράμμα τοποθετηθεί δίπλα σε ένα μόνο γράμμα για να σχηματιστεί μια νέα ομάδα. Όταν ένα γράμμα τοποθετείται ανάμεσα σε δύο μεμονωμένα γράμματα, έτσι ώστε να σχηματιστεί μια νέα ομάδα των τριών, τότε προστίθενται 3 πόντοι.

Το παράδειγμα δίνει αρχικά 5 βαθμούς. Για να λάβετε 11 βαθμούς, τα B, B και C πρέπει να τοποθετηθούν με τέτοιο τρόπο ώστε να δημιουργείται μια νέα ομάδα των δύο κάθε φορά. Μόνο δύο μονά B και δύο μονά C είναι διαθέσιμα για αυτό:



Υπάρχει μόνο μία δυνατότητα να σχηματιστούν τρεις νέες ομάδες των δύο με τους B, B και C χωρίς να μειωθεί το μέγεθος των υπάρχουσών ομάδων: να κάνουμε. Μόνο για την τελευταία πιθανότητα παίρνεις 11 πόντους, ενώ για τις υπόλοιπες 10 πόντους. Εδώ είναι και οι τέσσερις συνδυασμοί:



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Αυτό το παράδειγμα δείχνει ξεκάθαρα πώς προχωρούν οι επιστήμονες υπολογιστών όταν αναζητούν μια βέλτιστη λύση ή μια μέθοδο για την εξεύρεση λύσης.

Η ιδέα είναι να βρείτε μια τοποθέτηση των γραμμάτων B, B και C που να κερδίζει τους περισσότερους πόντους. Μια πρώτη προσέγγιση θα μπορούσε απλώς να είναι η δημιουργία όλων των δυνατοτήτων στις οποίες τρία γράμματα της σειράς

αντικαθίστανται από τα B, B και C. Για μια σειρά δώδεκα γραμμάτων, υπάρχουν  $12 * 11 * 10 = 1320$  δυνατότητες, για καθεμία από τις οποίες η βαθμολογία πρέπει να υπολογιστεί και να συγκριθεί για να προσδιοριστεί η καλύτερη λύση.

Ο αριθμός των λύσεων μπορεί να μειωθεί σημαντικά εάν λάβετε υπόψη ότι τα B, B και C βαθμολογούνται μόνο εάν βρίσκονται δίπλα στα ίδια γράμματα. Εάν λάβετε επίσης υπόψη ότι οι υπάρχουσες ομάδες δεν πρέπει να μειωθούν σε μέγεθος, ο αριθμός των λύσεων μειώνεται περαιτέρω. Εάν λάβετε επίσης υπόψη ότι η σύνδεση μεμονωμένων γραμμάτων φέρνει περισσότερους πόντους από τη μεγέθυνση υπάρχουσών ομάδων, τότε απομένει μόνο μία πιθανή τοποθέτηση.

Οι 1320 δυνατότητες που αναφέρονται παραπάνω δεν είναι πολλές για να υπολογίσει ένας υπολογιστής. Ωστόσο, η επιστήμη των υπολογιστών πάντα ρωτά πώς θα άλλαζε η προσπάθεια που απαιτείται για την επίλυση του προβλήματος καθώς μεγαλώνει το μέγεθος του προβλήματος. Με μεγαλύτερες σειρές και περισσότερα εναλλασσόμενα γράμματα, ο χώρος της λύσης θα ήταν αντίστοιχα μεγαλύτερος. Αξίζει τον κόπο να αναλύσουμε προσεκτικά το πρόβλημα και να αναπτύξουμε μια αποτελεσματική μέθοδο.

Από την άποψη της πληροφορικής, αυτό είναι ένα πρόβλημα βελτιστοποίησης. Στον τομέα της πληροφορικής, η βελτιστοποίηση παίζει καθοριστικό ρόλο. Έχει να κάνει με το να κάνουμε κάτι όσο το δυνατόν πιο αποτελεσματικό ή λειτουργικό εντός ορισμένων περιορισμών. Η βελτιστοποίηση αφορά την εύρεση της καλύτερης από όλες τις πιθανές λύσεις. Για παράδειγμα, όταν μιλάμε για βελτιστοποίηση κώδικα σε έναν μεταγλωττιστή, αναφερόμαστε στη διαδικασία τροποποίησης του κώδικα ώστε να γίνει πιο αποτελεσματικός και μικρότερος χωρίς να αλλάξει η έξοδος του. Αυτό είναι σημαντικό γιατί βοηθά τα προγράμματα να τρέχουν πιο γρήγορα και να χρησιμοποιούν λιγότερους πόρους. Ομοίως, η συμπίεση δεδομένων είναι ένα άλλο παράδειγμα βελτιστοποίησης. Περιλαμβάνει την κωδικοποίηση πληροφοριών χρησιμοποιώντας λιγότερα bit από την αρχική αναπαράσταση για εξοικονόμηση χώρου αποθήκευσης ή μείωση του όγκου των δεδομένων που πρέπει να μεταδοθούν. Σε αυτήν την εργασία, όταν προσπαθούμε να τακτοποιήσουμε βέλτιστα τα γράμματα «BBC», ουσιαστικά κάνουμε το ίδιο πράγμα - προσπαθούμε να βρούμε την πιο αποτελεσματική διάταξη που θα πετύχει τον στόχο μας, δηλαδή να δημιουργήσουμε την ακολουθία που μας δίνει το μεγαλύτερο σκορ. Για να γίνει αυτό, αποθηκεύουμε τα γράμματα του BBC για να αυξήσουμε το μήκος μιας υποακολουθίας των ίδιων γραμμάτων ή να δημιουργήσουμε μια νέα υποακολουθία. Για όλες τις ακολουθίες που δημιουργούμε με αυτόν τον τρόπο, υπολογίζουμε τις βαθμολογίες τους. Στη συνέχεια επιλέγουμε τη σειρά με την καλύτερη βαθμολογία. Όπως και στην πληροφορική, αναζητούμε την «καλύτερη» λύση εντός των περιορισμών που έχουμε θέσει.

### Υπολογιστική σκέψη

Όταν αναζητούμε μια λύση, χρησιμοποιούμε την αποσύνθεση, η οποία περιλαμβάνει τη διάσπαση του προβλήματος σε μικρότερα, πιο διαχειρίσιμα μέρη. Αυτή είναι μια βασική πτυχή της υπολογιστικής σκέψης. Το πρώτο βήμα σε αυτή τη διαδικασία είναι να προσδιορίσουμε όλες τις ακολουθίες που θα μπορούσαν ενδεχομένως να σχετίζονται με

τη λύση μας. Αφού έχουμε αυτές τις ακολουθίες, υπολογίζουμε μια βαθμολογία για κάθε μία. Αφού υπολογιστούν όλες οι βαθμολογίες, περνάμε στην αξιολόγηση.

Η αξιολόγηση είναι μια άλλη σημαντική πτυχή της υπολογιστικής σκέψης. Σε αυτήν την περίπτωση, αξιολογούμε τις βαθμολογίες των ακολουθιών μας. Τέλος, επιλέγουμε τη σειρά με την υψηλότερη βαθμολογία. Αυτή είναι η βέλτιστη λύση μας. Αξίζει να σημειωθεί ότι η αξιολόγηση περιλαμβάνει επίσης τον προσδιορισμό των ακολουθιών που αξίζουν να εξεταστούν αρχικά. Χρησιμοποιώντας την αποσύνθεση και την αξιολόγηση εργασιών, μπορούμε συστηματικά και αποτελεσματικά να βρούμε την καλύτερη λύση στο πρόβλημά μας.

## Χώρα προέλευσης θέματος

Σλοβακία

## 2024-CA-02 – Αλυσίδες λέξεων

Ηλικιακή ομάδα: Β' - Γ' Λυκείου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Αλυσίδες λέξεων

Ο Κάστορας παίζει με κωδικούς που αποτελούνται από τρεις χαρακτήρες. Τοποθετεί τους κωδικούς σε αλυσίδες κωδικών έτσι ώστε μόνο ένας χαρακτήρας να αλλάζει από τον ένα κωδικό στον επόμενο στην αλυσίδα κωδικών.

Για παράδειγμα, οι κωδικοί αυτοί μπορούν να τοποθετηθούν σε μια αλυσίδα κωδικών:  
XUG → XUD → XED → KED

Ο Κάστορας δημιούργησε εννέα κωδικούς: TOF, XEW, TEF, CET, COF, TEW, COT, CEF, και XEF.

Τους τοποθετεί σε αλυσίδες των τριών κωδικών έτσι ώστε κάθε ένας από τους εννέα κωδικούς να χρησιμοποιείται σε μία μόνο αλυσίδα κωδικών.

Καμία από τις παρακάτω αλυσίδες κώδικα δεν παραβιάζει τους κανόνες, αλλά μία από αυτές καθιστά αδύνατη την δημιουργία τριών αλυσίδων κώδικα χωρίς επανάληψη κωδικών.

### Ερώτηση

Ποια επιλογή δεν θα μπορούσε να είναι μία από τις αλυσίδες κώδικα του Κάστορα;

XEW → TEW → TEF

COF → COT → CET

TEF → CEF → COF

CEF → CET → COT

### Απάντηση

TEF → CEF → COF

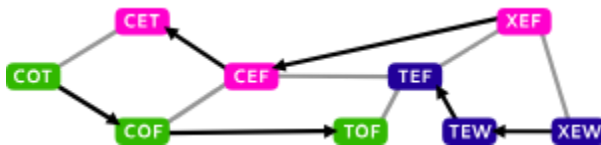
## Εξήγηση απάντησης

Η απάντηση είναι  $TEF \rightarrow CEF \rightarrow COF$ .

Για να λύσετε αυτό το πρόβλημα, είναι χρήσιμο να σχεδιάσετε ένα διάγραμμα. Σε αυτό το διάγραμμα, δύο λέξεις συνδέονται με μια γραμμή εάν μπορούν να διαδέχονται η μία την άλλη σε μια αλυσίδα λέξεων.



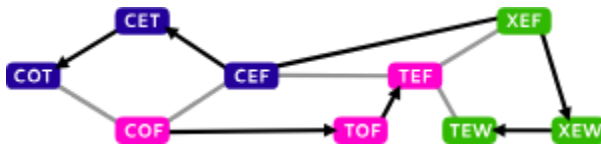
Η επιλογή  $XEW \rightarrow TEW \rightarrow TEF$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $XEF \rightarrow CEF \rightarrow CET$  και  $COT \rightarrow COF \rightarrow TOF$ , όπως φαίνεται.



Η επιλογή  $COF \rightarrow COT \rightarrow CET$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $CEF \rightarrow TEF \rightarrow TOF$  και  $XEF \rightarrow XEW \rightarrow TEW$ , όπως φαίνεται.

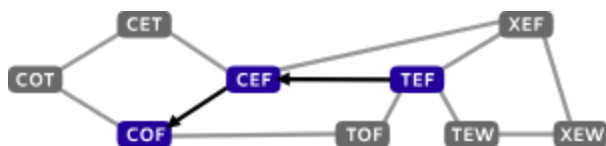


Η επιλογή  $CEF \rightarrow CET \rightarrow COT$  είναι μια πιθανή αλυσίδα λέξεων. Σε αυτήν την περίπτωση, οι άλλες αλυσίδες δύο λέξεων θα μπορούσαν να είναι  $COF \rightarrow TOF \rightarrow TEF$  και  $XEF \rightarrow XEW \rightarrow TEW$ , όπως φαίνεται.



Η επιλογή  $TEF \rightarrow CEF \rightarrow COF$  δεν είναι μια πιθανή αλυσίδα λέξεων επειδή, για παράδειγμα, στο διάγραμμα, το  $TOF$  συνδέεται μόνο με  $TEF$  και  $COF$ . Επομένως, το  $TOF$  πρέπει να είναι σε μια αλυσίδα λέξεων με τουλάχιστον μία από αυτές τις λέξεις. Δεδομένου ότι το  $TEF$  και το  $COF$  είναι και τα δύο στην αλυσίδα λέξεων για την τρίτη επιλογή, και το  $TOF$  δεν είναι, αυτό δεν αφήνει καμία πιθανή αλυσίδα λέξεων για το  $TOF$ .





## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Στα σχέδια με τα οποία έχουμε εξηγήσει τη λύση, κάθε λέξη αντιστοιχεί σε έναν κόμβο. Μια σύνδεση μεταξύ δύο κόμβων δείχνει ότι αυτές οι δύο λέξεις διαφέρουν ακριβώς σε μία γράμμα. Αυτοί οι δύο κόμβοι είναι τότε γείτονες.

Στην αλυσίδα λέξεων BAD — RAD — RAT — ROT, για παράδειγμα, το BAD και το RAD είναι γείτονες, αλλά το BAD και το RAT δεν είναι. Γενικά ονομάζουμε γράφους, δομές που αποτελούνται από κόμβους και ακμές και τις χρησιμοποιούμε για να χρησιμοποιήσουμε τις ακμές για να αναπαραστήσουμε τις σχέσεις μεταξύ των αντικειμένων που αντιπροσωπεύονται από τους κόμβους.

Η αλυσίδα λέξεων BAD — RAD — RAT περιλαμβάνει τρεις κόμβους και δύο άκρες. Μεταξύ BAD και RAT υπάρχει μια διαδρομή που συνδέει τους αντίστοιχους κόμβους μέσω του κοινού γείτονα RAD. Από καθεμία από τις εννέα λέξεις μπορούμε να φτάσουμε σε όλες τις άλλες λέξεις. Αυτό μπορεί να γίνει απευθείας (δηλ. εάν οι αντίστοιχοι κόμβοι είναι γείτονες) ή χρησιμοποιώντας μια μεγαλύτερη διαδρομή που εκτείνεται σε πολλούς γείτονες. Για να λύσουμε το πρόβλημα του κάστορα χρειαζόμαστε τρία τέτοια μονοπάτια. Κάθε κόμβος πρέπει να χρησιμοποιείται ακριβώς μία φορά.

Τα γραφήματα είναι ένα δημοφιλές εργαλείο που χρησιμοποιείται από επιστήμονες υπολογιστών για τη μοντελοποίηση πολύπλοκων ζητημάτων. Τα προγράμματα μπορούν να λύσουν πολλές εργασίες χρησιμοποιώντας γραφήματα.

Τα συστήματα πλοήγησης, για παράδειγμα, είναι προγράμματα που χρησιμοποιούν πολύ μεγάλα γραφήματα, βήμα προς βήμα από την αφετηρία μέχρι τον προορισμό.

## Χώρα προέλευσης θέματος

Καναδάς

## 2024-LT-05 – Φίλοι

Ηλικιακή ομάδα: Β' - Γ' Γυμνασίου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Φίλοι

Υπάρχουν επτά κάστορες σε μια τάξη με τα ονόματα: Alex, Bess, Cora, Dave, Eric, Fred, και Gigi. Μερικοί από τους κάστορες γνωρίζονται μεταξύ τους και άλλοι όχι.

Το παρακάτω διάγραμμα δείχνει ποιοι κάστορες γνωρίζονται μεταξύ τους με ένα πράσινο σημάδι.



	Alex	Bess	Cora	Dave	Eric	Fred	Gigi
Alex	■	✓	✓				
Bess	✓	■					✓
Cora	✓		■		✓		
Dave				■		✓	
Eric			✓		■		
Fred				✓		■	
Gigi		✓					■

Κάθε κάστορας έλαβε ένα διαφορετικό μήνυμα από κάποιον εκτός της τάξης και το μοιράστηκε με όλους τους κάστορες που γνωρίζει στην τάξη του. Κάθε φορά που ένας κάστορας λαμβάνει ένα νέο μήνυμα από κάποιον στην τάξη, το μοιράζεται αμέσως με όλους τους υπόλοιπους κάστορες που γνωρίζει στην τάξη.

### Ερώτηση

Ποιος ή ποιοι κάστορες στην τάξη έλαβαν τον λιγότερο αριθμό μηνυμάτων συνολικά;

### Απάντηση

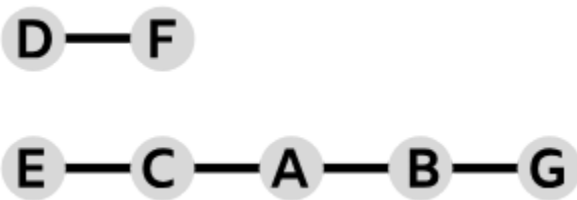
Dave, Fred

## Εξήγηση απάντησης

Η σωστή απάντηση είναι ο Dave και ο Fred.

Μπορούμε να χρησιμοποιήσουμε ένα διάγραμμα για να δείξουμε τις σχέσεις μεταξύ των καστόρων στην τάξη. Σε αυτό το διάγραμμα, κάθε κάστορας αντιπροσωπεύεται από έναν κύκλο, και αν οι δύο κάστορες γνωρίζονται μεταξύ τους, τότε οι κύκλοι τους συνδέονται με μια γραμμή.

Ο Dave γνωρίζει μόνο τον Fred και ο Fred μόνο τον Dave, επομένως συνδέουμε τους κύκλους τους μεταξύ τους και δεν τους συνδέουμε με άλλους κύκλους. Ο Eric γνωρίζει μόνο την Cora, οπότε συνδέουμε τους κύκλους τους. Η Cora γνωρίζει επίσης τον Alex, οπότε συνδέουμε τους κύκλους τους. Ο Άλεξ γνωρίζει επίσης την Bess, οπότε συνδέουμε τους κύκλους τους. Τέλος, η Bess γνωρίζει και την Gigi, οπότε συνδέουμε τους κύκλους τους.



Τώρα από το διάγραμμα, σαφώς καθένας από τους Dave και Fred μπορεί να λάβει μόνο 2 μηνύματα, ενώ ο Eric, η Cora, ο Alex, η Bess και η Gigi θα λάβει 5 μηνύματα.

Επομένως, ο Dave και ο Fred θα λάβουν τον λιγότερο αριθμό μηνυμάτων.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Οι γράφοι είναι ένα σημαντικό εργαλείο αφαίρεσης στην επιστήμη των υπολογιστών. Σε αυτήν την εργασία, ένα γράφημα παρουσιάζεται ως πίνακας γειτνίασης. Το πρόβλημα των φιλιών με κάστορες σχετίζεται με ένα πρόβλημα συνδεσιμότητας γράφου. Στη θεωρία γράφων και στην επιστήμη των υπολογιστών, ένας πίνακας γειτνίασης είναι ένας τετραγωνικός πίνακας που χρησιμοποιείται για να αναπαραστήσει ένα πεπερασμένο γράφο. Τα στοιχεία του πίνακα υποδεικνύουν εάν ζεύγη στοιχείων είναι γειτονικά ή όχι στο γράφημα. Ο γράφος σε αυτήν την εργασία είναι μη κατευθυνόμενος (δηλαδή όλες οι ακμές του είναι αμφίδρομες), επειδή ο πίνακας γειτνίασης είναι συμμετρικός.

Ο πίνακας γειτνίασης μπορεί να χρησιμοποιηθεί ως δομή δεδομένων για την αναπαράσταση γράφων σε προγράμματα υπολογιστών για το χειρισμό γραφημάτων. Η κύρια εναλλακτική δομή δεδομένων, που χρησιμοποιείται επίσης για αυτήν την εφαρμογή, είναι η λίστα γειτνίασης.

Χώρα προέλευσης θέματος

Λιθουανία

## 2024-IT-01b – Χάρτης φαγητού

Ηλικιακή ομάδα: Β' – Γ' Λυκείου

Βαθμός δυσκολίας: ΕΥΚΟΛΟ

Θέμα : Χάρτης φαγητού

Ο Κάστορας Θανάσης έκρυψε τις προμήθειες τροφής του κάτω από 9 από τα 17 δέντρα που περιβάλλουν μέρος της λίμνης. Ο Θανάσης έφτιαξε έναν χάρτη και έγραψε σε κάθε κουτί κατά μήκος της όχθης τον αριθμό των δέντρων που βρίσκονται κοντά του και έχουν κρυμμένη τροφή.

Για παράδειγμα, το κουτί «3» σημαίνει ότι ακριβώς 3 από τα 5 επισημασμένα δέντρα κοντά του έχουν κρυμμένη τροφή.



Αυτό δεν φαίνεται καλή ιδέα στη φίλη του, την Γιώτα και έχει δίκιο! Στην πραγματικότητα, ο Κάστορας Θανάσης θα μπορέσει να αναγνωρίσει με βεβαιότητα μόνο 7 από τα 9 δέντρα κάτω από τα οποία έκρυψε την τροφή.

### Ερώτηση

Επιλέξτε αυτά τα 7 δέντρα.  
Κάντε "Αποθήκευση" όταν τελειώσετε.

### Απάντηση

Η σωστή απάντηση είναι:



### Εξήγηση απάντησης

Τα επτά δέντρα κάτω από τα οποία σίγουρα κρύβεται φαγητό είναι αυτά που επισημαίνονται στην παρακάτω εικόνα.



Δύο δέντρα με κρυφή τροφή και ένα σίγουρα άδειο (X) μπορούν να αναγνωριστούν ξεκινώντας από το κάτω μέρος.



Από κάτω δεξιά προς τα αριστερά: το πρώτο δέντρο πρέπει να έχει τροφή για να ικανοποιήσει το χαμηλότερο 1. Το δεύτερο δέντρο πρέπει επίσης να έχει τροφή για να ικανοποιήσει το χαμηλότερο 2. Αντίθετα, αν υπήρχε τροφή κάτω από το δέντρο στην κάτω αριστερή γωνία, τότε δεν θα ήταν δυνατό να ικανοποιηθεί ο περιορισμός 2 που περικλείεται από τον μικρό κύκλο.

Με παρόμοιο σκεπτικό αποδεικνύεται εύκολα ότι, όσον αφορά το αριστερό μέρος της λιμνούλας, η συνέχεια επιβάλλεται:

		2	2	1
	2			
		3	2	1
				1

Όσον αφορά τη δεξιά πλευρά της λίμνης, υπάρχουν δύο δυνατότητες:

				?
1	1	2		
		2		
		2	?	
		1		

1	1	2	?	
		2		
		2		
		1	?	

Έτσι μόνο ένα ακόμα κόκκινο αστέρι μπορεί να προστεθεί.

Πώς μπορούμε να επισημοποιήσουμε καλύτερα το πρόβλημά μας; Αφού επιλέξουμε τα δύο κουτιά στην τελευταία σειρά που έχουν σίγουρα κρυμμένο φαγητό, δίνουμε ένα όνομα σε κάθε άλλο κουτί που περιέχει ένα δέντρο:

E	F	G	H	I	J	K
D	2	2	1	1	2	L
C	2				2	M
B	3	2	1		2	N
A	★ 🌲	★ 🌲	1		1	O

Οι αριθμοί στα μπλε πλαίσια εκφράζουν περιορισμούς. προχωρώντας δεξιόστροφα από κάτω, οι τρεις πρώτοι (1, 1, 2) είναι ήδη ικανοποιημένοι, για να ικανοποιηθεί ο περιορισμός 3 είναι απαραίτητο να επιλέξετε ακριβώς ένα ακόμη στοιχείο από το σύνολο {A, B, C}, και ούτω καθεξής. Τότε το πρόβλημα μπορεί να επισημοποιηθεί ως εξής:

- από τα {A, B, C} επιλέξτε ακριβώς ένα στοιχείο,
- από τα {B, C, D} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {C, D, E, F, G} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {F, G, H} επιλέξτε ακριβώς 2 στοιχεία,
- από {G, H, I} επιλέξτε ακριβώς ένα στοιχείο,
- από {H, I, J} επιλέξτε ακριβώς ένα στοιχείο,
- από {I, J, K, L, M} επιλέξτε ακριβώς 2 στοιχεία,
- από τα {L, M, N} επιλέξτε ακριβώς 2 στοιχεία,
- από {M, N, O} επιλέξτε ακριβώς 2 στοιχεία,
- από το {N, O} επιλέξτε ακριβώς ένα στοιχείο.

Από εξερεύνηση όλων των πιθανοτήτων (δεν μπορείτε να επιλέξετε το A από το πρώτο σετ, γιατί τότε θα πρέπει να επιλέξετε είτε το B είτε το C από το δεύτερο, αν επιλέξετε B από το πρώτο σετ, τότε πρέπει να επιλέξετε το D από το δεύτερο κ.ο.κ.) μπορεί να επαληθευτεί ότι υπάρχουν δύο λύσεις:

{B, D, F, H, K, M, N}, {B, D, F, H, L, M, O}.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής



Αυτή η εργασία είναι εμπνευσμένη από ένα διάσημο βιντεοπαιχνίδι παζλ λογικής, γνωστό με διαφορετικά ονόματα (Minesweeper, Flower Field), όπου παίζει και η τύχη. Και ακόμη και όταν ανοίγουν μεγάλες επιφάνειες, μπορεί να προκύψουν καταστάσεις όπου δεν υπάρχουν αρκετές διαθέσιμες πληροφορίες για συνεχίστε το παιχνίδι με ασφάλεια, οπότε στο τέλος πρέπει να βασιστείτε στην τύχη. Αυτή είναι ακριβώς η περίπτωση που προκύπτει στο έργο μας: η δεξιά πλευρά της λίμνης δεν μπορεί να λυθεί με ασφάλεια!

Σημειώστε ότι, γενικά, εάν υπάρχουν  $N$  δέντρα γύρω από τη λίμνη, αντιστοιχούν σε  $2^N$  πιθανές διαμορφώσεις, και τότε είναι συνήθως πολύ πιο εύκολο να ελέγξετε εάν μια δεδομένη λύση είναι συνεπής με τους περιορισμούς του προβλήματος παρά να την βρείτε!

Μέσω αυτής της εργασίας μπορούμε επίσης να δούμε πόσο συχνά ένα πρόβλημα στο σύνολό του μπορεί σταδιακά να αντιμετωπιστεί και να λυθεί σε ξεχωριστά μέρη. Μακράν η πιο ενδιαφέρουσα πτυχή από την άποψη της πληροφορικής είναι η επισημοποίηση του προβλήματος, μέσω μιας κατάλληλης δομής δεδομένων (όπως κάναμε για αυτό το πρόβλημα στο τελευταίο μέρος της εξήγησης), η οποία υποδηλώνει πώς, γενικά, αυτό το πρόβλημα είναι υπολογιστικά δύσκολο, καθώς και η αποδοχή πολλαπλών λύσεων.

### Υπολογιστική Σκέψη

Εάν δεν είστε πολύ άτυχοι, παίζοντας το Flower Field, η λογική παίζει σημαντικό ρόλο και η ερώτηση που τίθεται στο πρόβλημα μας το αποδεικνύει πειστικά: για να φτάσετε στη σωστή απάντηση συλλογίζοντας τους περιορισμούς, είναι απαραίτητο να διερευνήσετε μερικές διαφορετικές δυνατότητες και τις συνέπειες τους. Όμως, ορισμένα προβλήματα μπορεί να μην μπορούν να λυθούν με βεβαιότητα! Αυτός ο τύπος συλλογισμού είναι μεταξύ αυτών που χαρακτηρίζουν την υπολογιστική σκέψη. Ο λογικός συλλογισμός, η δημιουργία υποψήφιων λύσεων και η ικανοποίηση περιορισμών είναι κοινές διαδικασίες στην υπολογιστική σκέψη.

## Χώρα προέλευσης θέματος

Ιταλία

## 2024-CA-02 – Σφάλμα στην ζωγραφική

Ηλικιακή ομάδα: Β' – Γ' Λυκείου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Σφάλμα στην ζωγραφική

Το ρομπότ σχεδίασης του Στέφανου κατανοεί δύο βασικές οδηγίες κίνησης:

- Μ που σημαίνει “προχώρησε μπροστά 1 μονάδα”
- Δ που σημαίνει “στρίψε 90 μοίρες δεξιά”

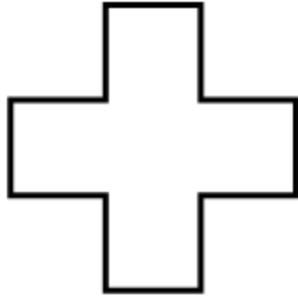
Επιπλέον, οποιαδήποτε άλλα γράμματα μπορούν να χρησιμοποιηθούν ως συντόμευση για πιο περίπλοκες ακολουθίες κινήσεων. Για παράδειγμα:

- A = MMM είναι συντόμευση κίνησης 3 μονάδων μπροστά.
- B = AΔA είναι συντόμευση για την εφαρμογή των κινήσεων του A (MMM), μίας στροφής 90 μοιρών δεξιά και ξανά την εφαρμογή του A.



Ένα πρόγραμμα μπορεί να έχει οποιονδήποτε αριθμό γραμμών κώδικα για τον ορισμό διαφορετικών συντομεύσεων. Η τελευταία γραμμή προορίζεται για την εντολή 'Ζωγράφισε' που με βάση τις συντομεύσεις θα παράξει το τελικό σχέδιο.

Ο Στέφανος θέλει να σχεδιάσει το ακόλουθο σχέδιο, όπου κάθε τμήμα γραμμής έχει μήκος 1 μονάδα.



Προγραμματίζει το ρομπότ του με αυτές τις τέσσερις γραμμές κώδικα:

1. Γραμμή 1: Π = ΜΔ
2. Γραμμή 2: Ρ = ΠΔΔ
3. Γραμμή 3: Σ = ΠΡΠΡ
4. Γραμμή 4: Ζωγράφισε ΣΣΣΣ

Δυστυχώς, το πρόγραμμα του Στέφανου έχει ένα τυπογραφικό λάθος και το ρομπότ αποτυγχάνει να παράγει το σωστό σχέδιο. Ακριβώς μία γραμμή κώδικα έχει ένα λανθασμένο σύμβολο.

## Ερώτηση

Ποια γραμμή κώδικα πρέπει να αλλάξει για να σχεδιαστεί σωστά το σχέδιο του Στέφανου;

## Απάντηση

Η γραμμή 3: Σ = ΠΡΠΡ

## Εξήγηση απάντησης

Ας υποθέσουμε ότι το ρομπότ είναι αρχικά στραμμένο προς τα δεξιά. Ας δούμε κάθε γραμμή κώδικα. Στη γραμμή 1, το ρομπότ κινείται μία μονάδα προς τα εμπρός και στη συνέχεια στρέφεται δεξιά.



Γραμμή 1: Π = ΜΔ

Στη γραμμή 2, το ρομπότ κινείται μία μονάδα προς τα εμπρός και στη συνέχεια στρίβει τρεις φορές δεξιά, που ουσιαστικά σημαίνει ότι στρίβει αριστερά. Αυτό φαίνεται παρακάτω:

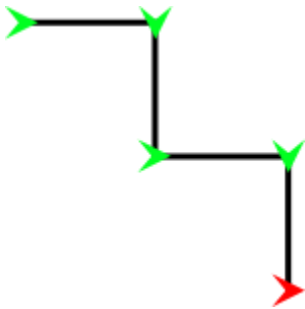
Γραμμή 1:  $P = \Pi$

$\Delta$   
 $\Delta$



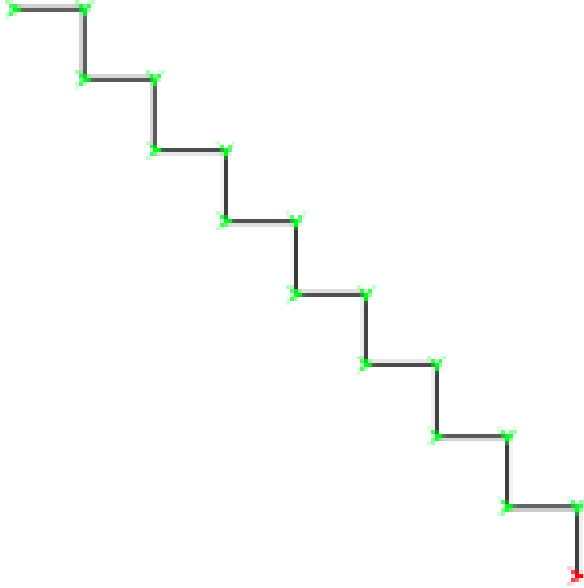
Στη γραμμή 3, χρησιμοποιούνται οι δύο προηγούμενες γραμμές, όπως φαίνεται παρακάτω:

Γραμμή 3:  $\Sigma = \Pi R \Pi R$



Στη γραμμή 4, η προηγούμενη γραμμή επαναλαμβάνεται τέσσερις φορές. Αυτό έχει ως αποτέλεσμα ένα διάγραμμα διαφορετικό από αυτό που ήθελε ο Στέφανος.

Γραμμή 4: Ζωγράφισε  $\Sigma \Sigma \Sigma \Sigma$



Στο σχέδιο που θέλει να σχεδιάσει ο Στέφανος υπάρχουν 12 γραμμές μήκους μίας μονάδας. Στη γραμμή 4, η προηγούμενη γραμμή επαναλαμβάνεται 4 φορές, πράγμα που σημαίνει ότι μία από αυτές τις επαναλήψεις σχεδιάζει το ένα τέταρτο του διαγράμματος-στόχου. Το ένα τέταρτο του συνολικού αριθμού των γραμμών που σχεδιάζονται είναι τρεις γραμμές, πράγμα που σημαίνει ότι το λάθος βρίσκεται στη γραμμή 3 η οποία σχεδιάζει συνολικά τέσσερις γραμμές.

Θα μπορούσε κανείς να σκεφτεί ότι είναι δυνατόν επίσης να διορθωθεί ο κώδικας αλλάζοντας τη γραμμή 1 σε  $\Pi = M$ , αλλά αυτό δεν ισχύει. Αν το αλλάξετε αυτό, θα επηρεάσει επίσης τον κώδικα για το  $\Sigma$  στη γραμμή 3 και μέσω αυτού θα οδηγηθούμε σε διαφορετικό σχέδιο.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Σε αυτή την εργασία, σας δίνεται μια νέα γλώσσα προγραμματισμού. Η γλώσσα αυτή εκτελεί εντολές διαδοχικά, αλλά σας επιτρέπει επίσης να ορίσετε διαδικασίες για ένα σύνολο εντολών. Η χρήση διαδικασιών ή συναρτήσεων σε ένα πρόγραμμα είναι κάτι πολύ συνηθισμένο. Τυπικά, ο ορισμός συναρτήσεων κάνει το πρόγραμμά σας πιο ευανάγνωστο. Σε αυτή την περίπτωση όμως, η σύμβαση ονοματοδοσίας των διαδικασιών στην πραγματικότητα κάνει το πρόγραμμά σας πιο δυσανάγνωστο.

Το πρώτο βήμα στην αποσφαλμάτωση αυτού του προγράμματος είναι να καταλάβετε τι κάνει, θα χρειαστεί να προσπαθήσετε να γίνετε ο ίδιος ο υπολογιστής και να προσομοιώσετε το πρόγραμμα στο μυαλό σας ή σε ένα κομμάτι χαρτί. Και στη συνέχεια, εφόσον γνωρίζετε ότι υπάρχει μόνο ένα λάθος, θα πρέπει να σκεφτείτε ποια γραμμή θα μπορούσε να αλλάξει για να διορθωθεί το πρόγραμμα.

## Υπολογιστική Σκέψη

Αποσύνθεση: Για την επίλυση αυτής της εργασίας, ο μαθητής χρησιμοποιεί την αποσύνθεση για να αναλύσει το πρόγραμμα από την ακολουθία των γραμμών. Στη συνέχεια, ο μαθητής αποσυνθέτει τις εντολές που αποτελούν κάθε γραμμή και προσδιορίζει τι θα σχεδίαζε το ρομπότ αν εκτελούνταν η γραμμή αυτή.

Τέλος, χρησιμοποιώντας αλγοριθμική σκέψη, ο μαθητής μοντελοποιεί αυτό που θα σχεδίαζε το ρομπότ για να ανακαλύψει πού βρίσκεται το σφάλμα. Μόλις εντοπιστεί το σφάλμα, τότε μπορεί να αξιολογήσει ποια γραμμή κώδικα περιέχει το σφάλμα.

## Χώρα προέλευσης θέματος

Αυστραλία

## 2024-FI-03 – Τοίχος από τούβλα

Ηλικιακή ομάδα: Β' – Γ' Λυκείου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Τοίχος από τούβλα

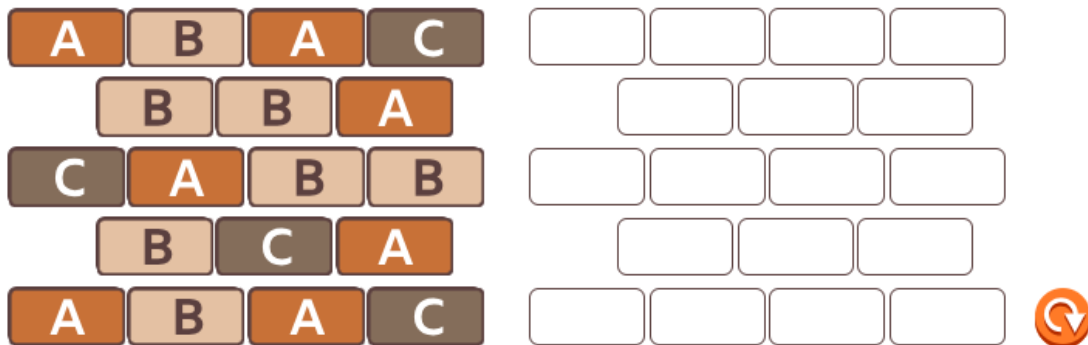
Ο Πέτρος έχτισε έναν τοίχο χρησιμοποιώντας τρία είδη τούβλων: Α, Β και C. Δυστυχώς, ο Πέτρος έχτισε τον τοίχο σε λάθος σημείο και πρέπει να τον μετακινήσει. Για να τα καταφέρει, μετακινεί ένα τούβλο την φορά ως εξής:

- Αφαιρεί κάθε τούβλο από τον αρχικό τοίχο που δεν έχει κανένα άλλο τούβλο ακριβώς από πάνω του.
- Τοποθετεί αυτό το τούβλο στον νέο τοίχο, είτε στο έδαφος είτε πάνω σε ένα από τα άλλα τούβλα, αλλά όχι κάτω από ήδη τοποθετημένα τούβλα.

Εφόσον ακολουθεί αυτούς τους κανόνες, ο Πέτρος δεν ενδιαφέρεται για τη σειρά με την οποία τα διάφορα είδη τούβλων μεταφέρονται από τον αρχικό τοίχο στον καινούργιο.

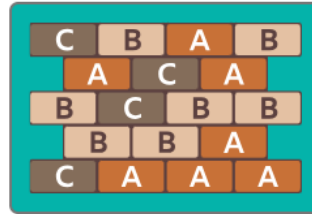
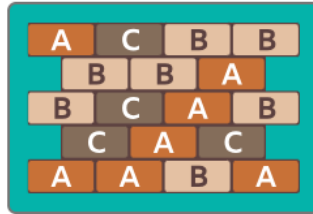
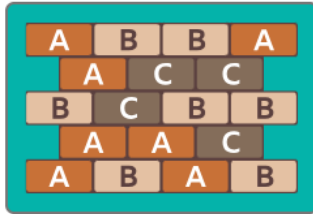
Στο παρακάτω σχέδιο φαίνεται ο αρχικός τοίχος του Πέτρου στα αριστερά. Μπορείτε να κάνετε κλικ και να σύρετε τούβλα για να χτίσετε έναν νέο τοίχο στα δεξιά - αλλά σημειώστε ότι έτσι δεν θα ελέγξετε αν έχετε ακολουθήσει σωστά τους κανόνες.

Για να αρχίσετε ξανά, πατήστε το κουμπί της επανεκκίνησης.



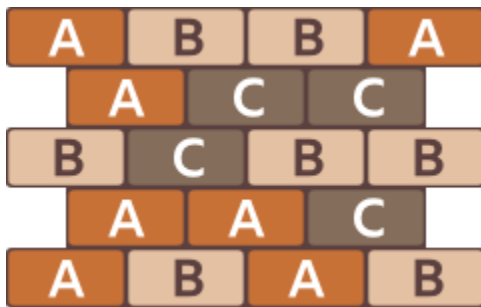
### Ερώτηση

Επιλέξτε τον τοίχο ή τους τοίχους που μπορεί να χτίσει ο Πέτρος ακολουθώντας τους κανόνες.



## Απάντηση

Μόνο οι δύο παρακάτω τοίχοι μπορούν να χτιστούν:



## Εξήγηση απάντησης

Για αυτήν την εξήγηση θα ονομάσουμε τους τοίχους A, B, Γ, Δ όπως φαίνεται παρακάτω:

Τοίχος A

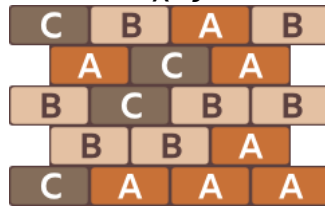


Τοίχος B





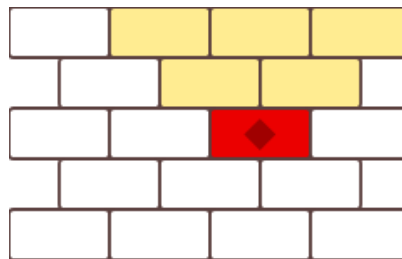
Τοίχος Γ



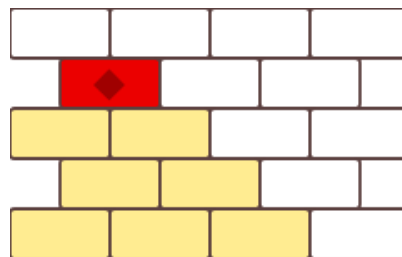
Τοίχος Δ



Παρατηρήστε το διάγραμμα στα αριστερά, όπου φαίνεται ένα τούβλο με κόκκινο χρώμα στον αρχικό τοίχο. Είναι η κορυφή ενός ανάποδου τριγώνου που εκτείνεται προς τα πάνω - τα τούβλα που επισημαίνονται με κίτρινο χρώμα είναι αυτά που πρέπει πρώτα να αφαιρεθούν πριν να μπορεί να μετακινηθεί το κόκκινο τούβλο. Ομοίως, το διάγραμμα στα δεξιά δείχνει ένα κόκκινο τούβλο που είναι η κορυφή ενός κανονικού τριγώνου που εκτείνεται προς τα κάτω (κομμένο στο πλάι σε αυτή την περίπτωση) - όλα τα κίτρινα τούβλα πρέπει να τοποθετηθούν πριν μπορέσει να τοποθετηθεί το κόκκινο τούβλο.



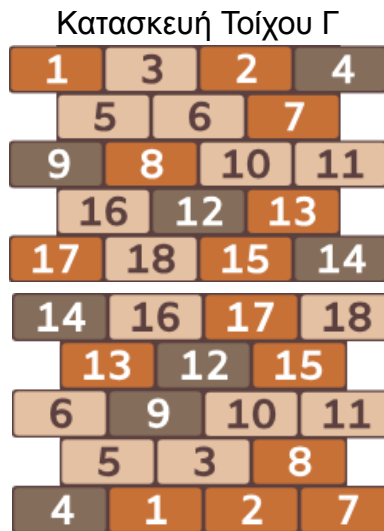
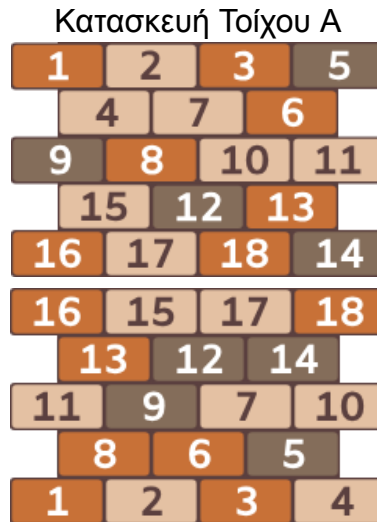
Αρχικός τοίχος



Νέος τοίχος

Χρησιμοποιώντας αυτή την ιδέα του τριγώνου, μπορούμε να επιβεβαιώσουμε ότι οι τοίχοι A και Γ μπορούν να χτιστούν. Παρακάτω δείχνουμε μια πιθανή κατασκευή για τον καθένα - υπάρχουν πολλές άλλες. Τα τούβλα στα παρακάτω διαγράμματα αριθμούνται σύμφωνα με τη σειρά με την οποία μετακινούνται από τον αρχικό τοίχο στον νέο τοίχο. Για να ελέγξουμε ότι οι κανόνες έχουν ακολουθηθεί σωστά, παρατηρούμε τα εξής:

- για κάθε τούβλο στον αρχικό τοίχο, κάθε άλλο τούβλο στο ανάποδο τρίγωνο του έχει μικρότερο αριθμό
- για κάθε τούβλο στον αρχικό τοίχο, κάθε άλλο τούβλο στο κανονικό τρίγωνο έχει μικρότερο αριθμό.



Το παραπάνω διάγραμμα δείχνει γιατί οι τοίχοι Β και Δ δεν μπορούν να χτιστούν σύμφωνα με τους κανόνες.

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Το πρόβλημα σχετίζεται με την ανάλυση των εξαρτήσεων (ή περιορισμών προτεραιότητας) μεταξύ αντικειμένων. Πρόκειται για ένα κλασικό πρόβλημα στην επιστήμη των υπολογιστών. Οι εξαρτήσεις μεταξύ αντικειμένων - σε αυτή την περίπτωση, μεταξύ των τούβλων - εκφράζονται συνήθως με τη μορφή ενός γραφήματος εξαρτήσεων. Μια νόμιμη σειρά επεξεργασίας, σε αυτό το θέμα, η σειρά με την οποία μπορούν να αφαιρεθούν και να τοποθετηθούν τούβλα - αντιστοιχεί σε μια τοπολογική διάταξη των αντικειμένων.

### Υπολογιστική σκέψη

Αυτή η εργασία συνδυάζει δύο σημαντικές δεξιότητες: τη συστηματική τήρηση κανόνων για να διαπιστωθεί ότι υπάρχει λύση για ορισμένες περιπτώσεις και την προσεκτική κατασκευή ενός λογικού επιχειρήματος για να διαπιστωθεί ότι δεν υπάρχει λύση για άλλες περιπτώσεις. Το τελευταίο είναι συχνά πιο δύσκολο. Στην παρεχόμενη εξήγηση, οι δύο κανόνες που ορίζουν τη μετακίνηση των τούβλων αντιμετωπίζουν ο ένας τον άλλο για να παραχθούν δύο αντιφατικές καταστάσεις. Μια πιο εξειδικευμένη προσέγγιση είναι η ανάλυση ενός δέντρου αποφάσεων, δοκιμάζοντας τις συνέπειες αυθαίρετων επιλογών που τηρούν τους κανόνες μέχρι να αποδειχθεί ότι όλα τα κλαδιά είναι αδιέξοδα. Αν και είναι δυσκολότερο να κατασκευάσει κανείς μια πειστική και συνοπτική εξήγηση ότι δεν υπάρχει λύση χρησιμοποιώντας αυτή την προσέγγιση, αποτελεί φυσικό σημείο εκκίνησης για τη διερεύνηση τέτοιων προβλημάτων.

## Χώρα προέλευσης θέματος

Φιλανδία

## 2024-PK-03 – Σιδηροδρομικό δίκτυο

Ηλικιακή ομάδα: Β' - Γ' Λυκείου

Βαθμός δυσκολίας: ΜΕΤΡΙΟ

Θέμα : Σιδηροδρομικό δίκτυο

Στη χώρα της Bebrania, οι γειτονικοί οικισμοί συνδέονται με ένα δίκτυο σιδηροδρομικών γραμμών. Για κάθε γραμμή, υπάρχει ένας περιορισμένος αριθμός τρένων που μπορούν να ταξιδεύουν κατά μήκος της κάθε μέρα, όπως φαίνεται στο παρακάτω διάγραμμα. Διαφορετικά τρένα μπορεί να κινούνται στην ίδια γραμμή.



Η πόλη Q προσφέρθηκε να στείλει υλικά στην πόλη P. Προς αυτή την κατεύθυνση, τα τρένα πρέπει πάντα να ακολουθούν τα βέλη.

### Ερώτηση

Ποιος είναι ο μέγιστος αριθμός τρένων που μπορούν να αναχωρούν από την πόλη Q και να φτάνουν στην πόλη P κάθε μέρα;

13

15

19

22

### Απάντηση

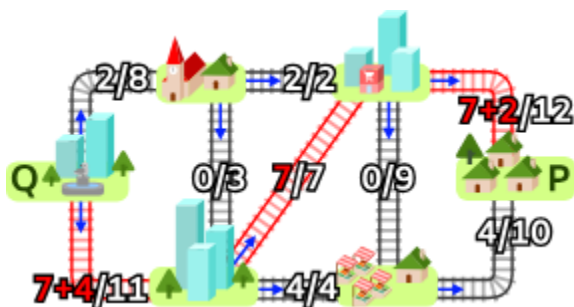
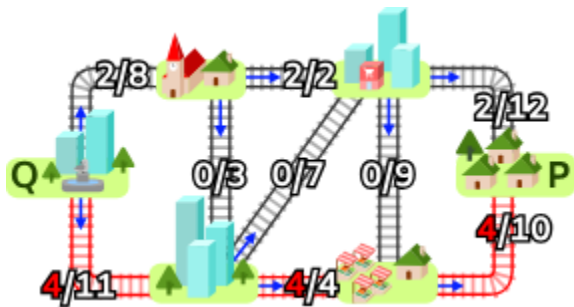
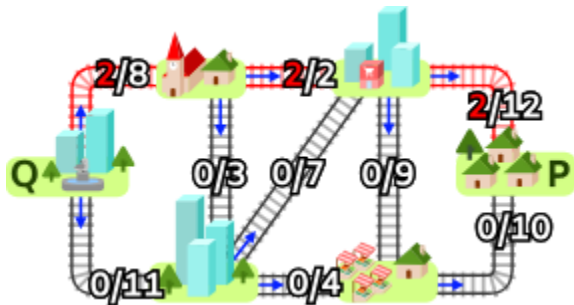
13

## Εξήγηση απάντησης

Η σωστή απάντηση είναι 13.

Η πρώτη παρατήρηση είναι ότι η πόλη Q δεν μπορεί να στείλει στην πόλη P περισσότερα από  $8+11=19$  τρένα, ο αριθμός των τρένων που μπορούν να φύγουν από την πόλη Q. Επομένως, η απάντηση "22" είναι λανθασμένη.

Ένας τρόπος για να λυθεί αυτό το πρόβλημα είναι να βρείτε μια συντομότερη δυνατή διαδρομή από την πόλη Q στην πόλη P (στην περίπτωση αυτή, μήκους 3) και να καθορίσετε τον μεγαλύτερο αριθμό τρένων που μπορεί να φιλοξενήσει. Στη συνέχεια επαναλαμβάνουμε αυτό το βήμα στο υπόλοιπο δίκτυο με τον υπόλοιπο αριθμό τρένων που είναι διαθέσιμα σε κάθε γραμμή. Τα παρακάτω διαγράμματα δείχνουν τρία μονοπάτια που βρέθηκαν με αυτόν τον τρόπο, με συνδυασμένη χωρητικότητα  $2+4+7=13$  τρένα. Υπάρχουν άλλοι τρόποι για να στείλετε 13 τρένα από την πόλη Q στην πόλη P, συμπεριλαμβανομένων ορισμένων που χρησιμοποιούν μεγαλύτερα μονοπάτια.



Για να ελέγξετε ότι το 13 είναι το μέγιστο, παρατηρήστε τα τρία ίχνη στη μέση του δικτύου με χωρητικότητες 2, 7 και 4: όλα τα μονοπάτια περνούν από ένα από αυτά, επομένως δεν μπορούμε να κάνουμε καλύτερα από  $2+7+4=13$ .

## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Το πρόβλημα σε αυτήν την εργασία είναι ένα παράδειγμα του προβλήματος μέγιστης ροής. Προβλήματα αυτού του τύπου είναι σαν ένα παιχνίδι παζλ με το νερό να ρέει μέσα από σωλήνες: έχετε ένα δίκτυο σωλήνων με διαφορετικές χωρητικότητες και θέλετε να υπολογίσετε πόσο νερό μπορεί να ρέει από μια πηγή σε μία έξοδο.

Το πρόβλημα μέγιστης ροής μπορεί να λυθεί χρησιμοποιώντας διάφορους αλγόριθμους: Αλγόριθμος Ford-Fulkerson, Αλγόριθμος Edmonds-Karp, Αλγόριθμος Dinic και πολλοί άλλοι. Τα αναπόφευκτα κομμάτια στη μέση του δικτύου σε αυτήν την εργασία σχηματίζουν αυτό που είναι γνωστό ως «ελάχιστη περικοπή».

Καθένας από τους παραπάνω αλγόριθμους έχει τα πλεονεκτήματα και τα μειονεκτήματά του όσον αφορά την απόδοση χρόνου εκτέλεσης, την ευκολία εφαρμογής του και την εφαρμογή σε διαφορετικούς τύπους δικτύων. Η επιλογή του αλγορίθμου εξαρτάται από παράγοντες όπως το μέγεθος του δικτύου, η φύση των χωρητικότητων και το επιθυμητό επίπεδο βελτιστοποίησης.

Οι αλγόριθμοι μέγιστης ροής χρησιμοποιούνται στη βελτιστοποίηση της κυκλοφορίας, στα συστήματα διανομής (π.χ. νερό και ηλεκτρική ενέργεια), στην κατανομή πόρων, στη δικτύωση υπολογιστών και πολλά άλλα.

### Υπολογιστική σκέψη

Η επίλυση αυτής της εργασίας περιλαμβάνει αλγοριθμική σκέψη και αξιολόγηση. Η λύση του προβλήματος αποσυντίθεται σε μια ακολουθία βημάτων, καθένα από τα οποία αποτελείται από την εύρεση μιας διαδρομής μεταξύ Q και P που έχει μέγιστη χωρητικότητα. Ο αλγόριθμος τερματίζεται όταν δεν υπάρχει άλλη διαθέσιμη τέτοια διαδρομή.

## Χώρα προέλευσης θέματος

Πακιστάν

## 2024-AU-01 – Κρυπτογραφημένη εικόνα

Ηλικιακή ομάδα: Β' – Γ' Λυκείου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Κρυπτογραφημένη εικόνα

Ο Στέργιος επινόησε μια νέα μέθοδο για την κρυπτογράφηση εικόνων χρησιμοποιώντας τις πράξεις "H" (οριζόντια) και "V" (κάθετα).

Μια εικόνα είναι ουσιαστικά ένα ορθογώνιο που χωρίζεται σε σειρές και στήλες τετραγώνων που ονομάζονται εικονοστοιχεία (pixels), με κάθε εικονοστοιχείο να αποθηκεύει ένα χρώμα.

Σε κάθε εφαρμογή της πράξης "H":

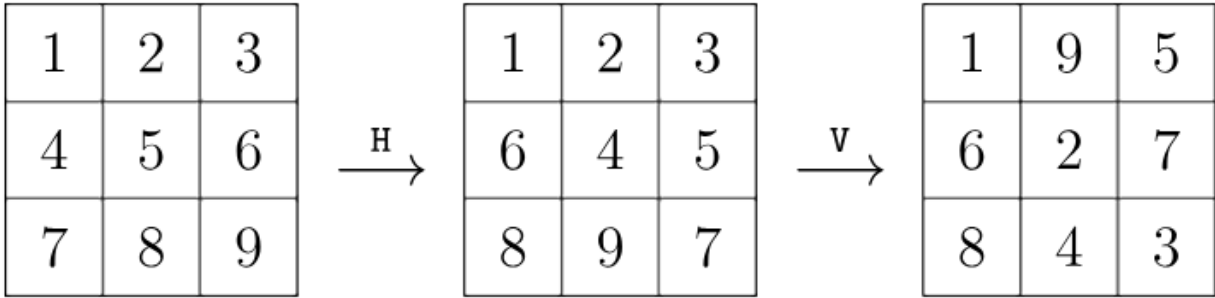
- Κάθε εικονοστοιχείο στην 1η σειρά παραμένει στη θέση του (δηλαδή δεν μετακινείται).
- Κάθε εικονοστοιχείο στην 2η σειρά μετακινείται 1 θέση προς τα δεξιά.
- Κάθε εικονοστοιχείο στην 3η σειρά μετακινείται 2 θέσεις προς τα δεξιά.
- Κάθε εικονοστοιχείο στην N-οστή σειρά μετακινείται  $N - 1$  θέσεις προς τα δεξιά, όπου  $N$  ο αριθμός της σειράς.

Όταν τα εικονοστοιχεία σε οποιαδήποτε σειρά σπρώχνονται έξω από το δεξιό άκρο της εικόνας, διατηρούνται στη σειρά και μετακινούνται στην αρχή του αριστερού άκρου.

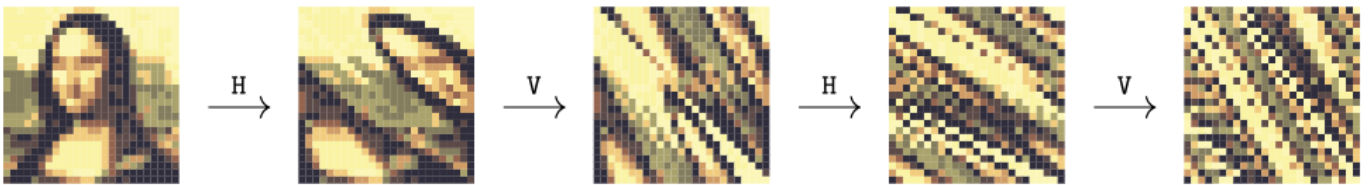
Ομοίως, σε κάθε εφαρμογή της πράξης "V":

- Κάθε εικονοστοιχείο στην N-οστή στήλης μετακινείται  $N - 1$  θέσεις προς τα κάτω και τα εικονοστοιχεία που σπρώχνονται έξω από το κάτω άκρο της εικόνας επιστρέφουν μετακινούνται στην αρχή του πάνω άκρου.

Ακολουθεί ένα παράδειγμα μιας εικόνας 3 επί 3 με τα χρώματα να φαίνονται ως αριθμοί από το 1 έως το 9:

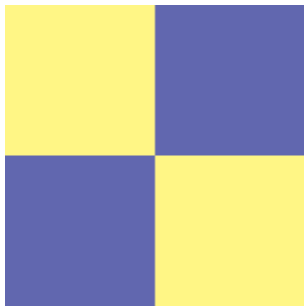


Να πώς μπορεί να χρησιμοποιηθεί η ακολουθία "HVHV" για την κρυπτογράφηση μιας εικόνας 25 επί 25 της Μόνα Λίζα:



### Ερώτηση

Ο Στέργιος κρυπτογραφεί την ακόλουθη εικόνα 1000 επί 1000 εφαρμόζοντας την πράξη "V" και μετά την πράξη "H".



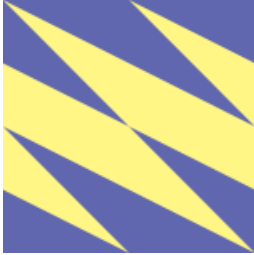
Ποια από τις παρακάτω επιλογές ταιριάζει με την κρυπτογραφημένη εικόνα;





## Απάντηση

Η σωστή απάντηση είναι:



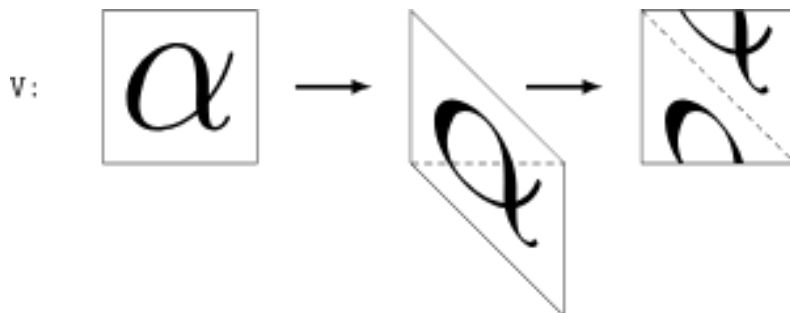
## Εξήγηση απάντησης

Η εικόνα έχει υψηλή ανάλυση, οπότε η κίνηση των μεμονωμένων εικονοστοιχείων μπορεί να προσεγγιστεί αρκετά καλά από τους γεωμετρικούς μετασχηματισμούς των σχημάτων που αναπαριστώνται στην εικόνα.

Στο παράδειγμα 3 x 3, βλέπουμε ότι η πρώτη στήλη (1, 4, 7) γίνεται η διαγώνιος από πάνω αριστερά προς κάτω δεξιά, ενώ η διαγώνιος από πάνω δεξιά προς κάτω αριστερά (3, 5, 7) γίνεται η τελευταία στήλη.

Αυτό υποδηλώνει ότι λαμβάνει χώρα μια διάτμηση, η οποία φαίνεται καλύτερα από την πρώτη εικόνα στο παράδειγμα της Μόνα Λίζα. Θα αναλύσουμε τις γεωμετρικές επιδράσεις των δύο πράξεων λεπτομερέστερα, ξεκινώντας με την  $V$ , δεδομένου ότι θα εφαρμοστεί πρώτη.

Το αποτέλεσμα της  $V$  μπορεί να χωριστεί σε δύο στάδια. Πρώτον, διατμήστε το ορθογώνιο κάθετα προς τα κάτω δεξιά για να σχηματίσετε ένα παραλληλόγραμμο με παραμορφωμένη εικόνα. Στη συνέχεια, μετατοπίστε κάθετα το προεξέχον τρίγωνο στο κάτω μέρος για να σχηματίσετε μια νέα ορθογώνια εικόνα του αρχικού μεγέθους:



Ομοίως, το αποτέλεσμα της  $H$  είναι μια οριζόντια διάτμηση που ακολουθείται από μια μερική οριζόντια μετατόπιση:



Εφαρμόζοντας αυτά τα βήματα στη δεδομένη εικόνα, έχουμε την παρακάτω ακολουθία:



## Σύνδεση με την Υπολογιστική Σκέψη

### Επιστήμη της πληροφορικής

Κατά τη μεταφορά πληροφοριών μπορεί να είναι σημαντική η κρυπτογράφηση των δεδομένων, έτσι ώστε κάποιος που υποκλέπτει τις πληροφορίες σας να μην μπορεί εύκολα να ανακαλύψει ποια ήταν τα αρχικά δεδομένα. Σε αυτή την εργασία σας έχει δοθεί ένας αλγόριθμος κρυπτογράφησης που φαίνεται αρκετά εύκολος, είναι εύκολο να αντιστραφεί, αλλά εξακολουθεί να παράγει μια νέα εικόνα που δεν καθιστά αμέσως σαφές ποια ήταν η αρχική. Στην πραγματικότητα, αν δεν γνωρίζετε τον μηχανισμό κρυπτογράφησης θα μπορούσαν να υπάρχουν πολλές πιθανές αρχικές εικόνες. Σημειώστε όμως ότι δεν υπάρχει μυστικό κλειδί, οπότε το μόνο που χρειάζεται να γνωρίζετε είναι ο αλγόριθμος κρυπτογράφησης και θα μπορούσατε να αποκρυπτογραφήσετε οποιαδήποτε εικόνα που έχει κρυπτογραφηθεί με αυτόν τον τρόπο.

### Υπολογιστική σκέψη

Η διαδικασία της αφαίρεσης συχνά μας επιτρέπει να κατανοήσουμε ένα πρόβλημα ή έναν αλγόριθμο με πιο ευρετικά μέσα - κατανοώντας την «ουσία» της διαδικασίας, αντί να απορροφήσουμε όλες τις λεπτές λεπτομέρειες. Για παράδειγμα, η απόσταση ενός αλγορίθμου στις βασικές αριθμητικές του ιδιότητες μπορεί να μας επιτρέψει να ελέγξουμε τη σχέση μεταξύ ζευγών εισόδων και εξόδων χωρίς απαραίτητα να κάνουμε μια πλήρη ανάλυση. Η καινοτομία σε αυτό το πρόβλημα είναι ότι η ευρετική προσέγγιση βασίζεται στη γεωμετρία και όχι στην άλγεβρα.

Χώρα προέλευσης θέματος

Αυστραλία

## 2024-DE-06a – Μηχανή μπαλονιών

Ηλικιακή ομάδα: Β' – Γ' Λυκείου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

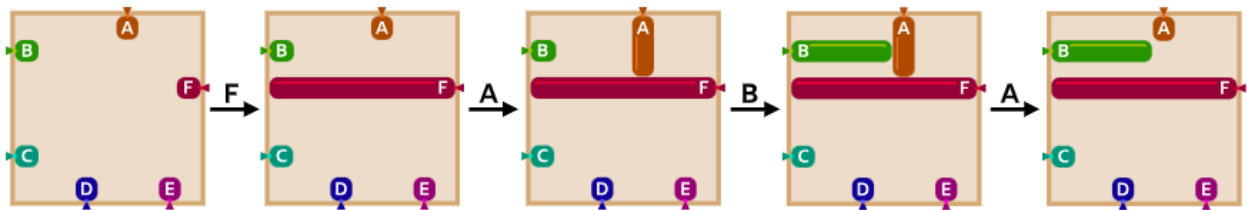
Θέμα : Μηχανή μπαλονιών

Το μηχάνημα διαβάζει γράμματα ένα-ένα. Όταν διαβάζει ένα γράμμα:

Εάν το μπαλόνι που φέρει την ετικέτα με αυτό το γράμμα είναι ξεφουσκωτό, φουσκώνει μέχρι να αγγίξει ένα άλλο μπαλόνι ή την απέναντι άκρη του πλαισίου.

Διαφορετικά ξεφουσκώνει το μπαλόνι που φέρει την ετικέτα με αυτό το γράμμα.

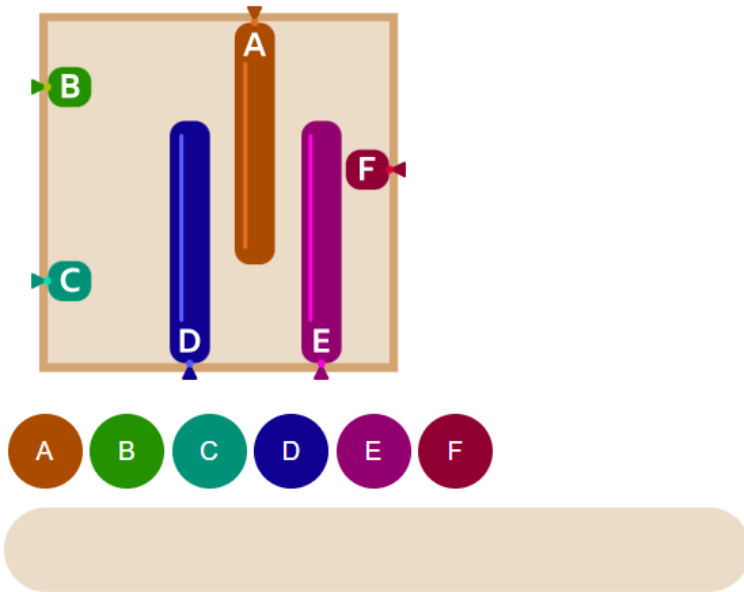
Για παράδειγμα, εάν όλα τα μπαλόνια είναι ξεφουσκωτά στην αρχή και το μηχάνημα γράφει F, A, B και μετά A, θα κάνει τα εξής:



### Ερώτηση

Για να ξεκινήσετε, κάθε μπαλόνι ξεφουσκώνεται και στη συνέχεια το μηχάνημα διαβάζει μια ακολουθία εννέα γραμμάτων. Το αποτέλεσμα είναι η παρακάτω εικόνα. Εισαγάγετε την ακολουθία των εννέα γραμμάτων.

Σύρετε τα γράμματα στη γραμμή για να σχηματίσετε την ακολουθία.



### Απάντηση

BEBCACBDB

BECBACBDB

BEBCABCDB

BECBABCDB

### Εξήγηση απάντησης

Υπάρχουν 4 πιθανές σωστές απαντήσεις με εννέα γράμματα:

BEBCACBDB

BECBACBDB

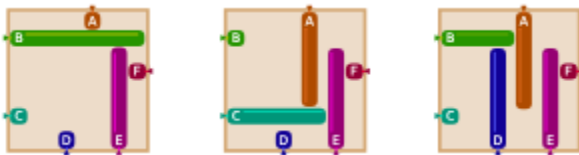
BEBCABCDB

BECBABCDB

Δείχνουμε λεπτομερή εξήγηση για μία απάντηση: B E B C A C B D B. Οι παρακάτω εικόνες απεικονίζουν την εκτέλεση αυτών των οδηγιών. Δείχνουν ενδιάμεσες καταστάσεις. Η πρώτη εικόνα δείχνει την κατάσταση μετά την εκτέλεση του B E. Μπορείτε να δείτε ότι είναι απαραίτητο να διογκωθεί το B πριν από το E για να γίνει το B κατάλληλο φράγμα για το φούσκωμα του E.

Η δεύτερη εικόνα δείχνει την κατάσταση μετά το B E B C A. Το μπαλόνι C χρησιμεύει ως φράγμα για το A και επομένως πρέπει να φουσκώσει πριν από το A. Επιπλέον το μπαλόνι B πρέπει να ξεφουσκώσει, πριν μπορέσει να φουσκώσει το A, διαφορετικά δεν θα φτάσει στο απαιτούμενο μήκος.

Η τρίτη εικόνα δείχνει την κατάσταση μετά το B E B C A C B D. Το B χρησιμεύει ως φράγμα για το D και το C πρέπει να ξεφουσκώσει πριν φουσκώσει το D.



Για να βρείτε όλες τις σωστές απαντήσεις, είναι χρήσιμο να αναπαραστήσετε το πρόβλημα χρησιμοποιώντας ένα κατευθυνόμενο γράφημα. Τα γράμματα αντιπροσωπεύουν τα μπαλόνια. Ένα βέλος από το B στο E υποδεικνύει ότι το μπαλόνι B πρέπει να φουσκώσει πριν από το μπαλόνι E κ.λπ. Υπάρχουν δύο περιπτώσεις όπου η σειρά των B και C μπορεί να ποικίλλει (υποδεικνύεται με διακεκομμένο ορθογώνιο).

Κάθε σωστή απάντηση (ακολουθία μπαλονιών που θα φουσκώσουν) είναι μια τοπολογική σειρά αυτού του γραφήματος. Σε αυτή την περίπτωση έχουμε 4 τοπολογικές ταξινομήσεις οπότε υπάρχουν τέσσερις πιθανές σωστές λύσεις.



Παρατήρηση για την αξιολόγηση της ορθότητας της λύσης σας: τα φουσκωμένα μπαλόνια πρέπει να εμφανίζονται περιττοί χρόνοι → A, D και E πρέπει να εμφανίζονται περιττοί χρόνοι. Τα ξεφουσκωμένα μπαλόνια πρέπει να εμφανίζονται ζυγές φορές (συμπεριλαμβανομένου του 0) → B, C και F.

## Σύνδεση με την Υπολογιστική Σκέψη

Επιστήμη της πληροφορικής

Η ακολουθία των γραμμάτων είναι ένα πρόγραμμα υπολογιστή, το οποίο ελέγχει μια μηχανή. Κάθε γράμμα είναι μια δήλωση που κάνει το μηχάνημα να φουσκώνει ή να ξεφουσκώνει ένα μπαλόνι. Όπως στα περισσότερα προγράμματα υπολογιστών αυτού του τύπου, η σειρά των δηλώσεων είναι απαραίτητη. Για παράδειγμα, η ακολουθία B E κάνει το μηχάνημα να δημιουργήσει μια διαφορετική εικόνα από την ακολουθία E B.

Ο κατευθυνόμενος γράφος που χρησιμοποιείται στην εξήγηση της απάντησης δεν έχει κύκλους και επομένως ονομάζεται κατευθυνόμενο άκυκλο γράφημα (DAG). Οποιοδήποτε DAG έχει τουλάχιστον μία τοπολογική διάταξη και είναι γνωστοί γρήγοροι αλγόριθμοι για την κατασκευή μιας τοπολογικής ταξινόμησης. Τα DAG έχουν πολυάριθμες εφαρμογές στην πληροφορική, π.χ. προγραμματισμός εργασιών.

## Χώρα προέλευσης θέματος

Γερμανία

## 2024-HU-04 – Palago

Ηλικιακή ομάδα: Β' – Γ' Λυκείου

Βαθμός δυσκολίας: ΔΥΣΚΟΛΟ

Θέμα : Palago

Το Palago είναι ένα παιχνίδι δύο παικτών που παίζεται με ίδια εξαγωνικά κομμάτια. Κάθε κομμάτι έχει μπλε και λευκό χρώμα όπως φαίνεται στην εικόνα. Ο ένας παίκτης είναι λευκός, ο άλλος μπλε. Στόχος του παιχνιδιού είναι οι παίκτες να σχηματίσουν ένα κλειστό σχήμα του χρώματός τους.



Ο πρώτος παίκτης τοποθετεί δύο κομμάτια το ένα δίπλα στο άλλο, τα χρώματα των οποίων πρέπει να ταιριάζουν στο σημείο επαφής τους. Στη συνέχεια, οι παίκτες τοποθετούν εναλλάξ δύο κομμάτια ο καθένας, τηρώντας τους ακόλουθους κανόνες:

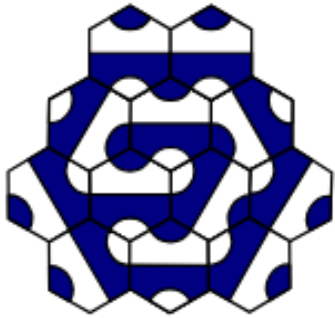
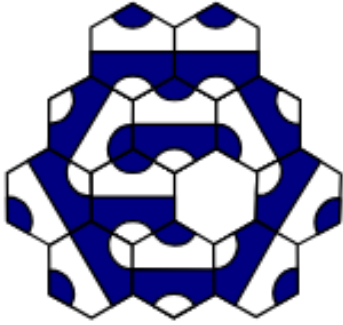
1. Τουλάχιστον ένα από τα δύο κομμάτια πρέπει να τοποθετηθεί δίπλα σε ένα κομμάτι που έχει ήδη παιχτεί.
2. Τα δύο κομμάτια που τοποθετούνται πρέπει να είναι το ένα δίπλα στο άλλο, δηλαδή να έχουν ένα κοινό σημείο επαφής.
3. Όλα τα χρώματα μεταξύ των σημείων επαφής δύο κομματιών πρέπει να ταιριάζουν.

Ένας παίκτης κερδίζει αν μπορεί να φτιάξει ένα κλειστό σχήμα του χρώματός του με τουλάχιστον μία ευθεία πλευρά. Ωστόσο, χάνει αν φτιάξει ένα κλειστό σχήμα και των δύο χρωμάτων στον γύρο του. Αν χρειάζεται μόνο ένα κομμάτι για να τελειώσει το παιχνίδι, το δεύτερο δεν χρειάζεται να τοποθετηθεί.

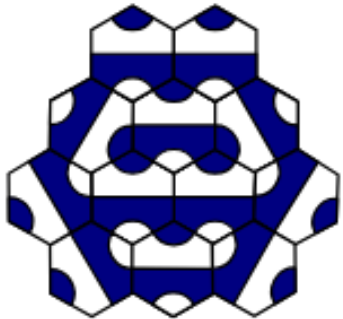
Η τωρινή κατάσταση του παιχνιδιού

Αφού τοποθετηθεί το επόμενο κομμάτι





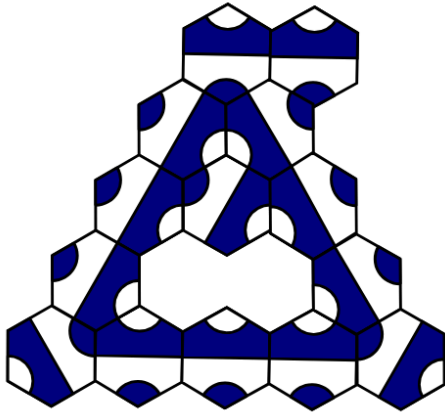
Ο λευκός παίκτης κερδίζει επειδή έχει σχηματίσει ένα κλειστό λευκό σχήμα με μια ευθεία πλευρά.



Ο λευκός παίκτης χάνει επειδή έχει σχηματίσει όχι μόνο ένα λευκό κλειστό σχήμα αλλά και ένα μπλε κλειστό σχήμα με μια ευθεία πλευρά.

## Ερώτηση

Ποιος παίκτης θα κερδίσει στην ακόλουθη κατάσταση παιχνιδιού εάν και οι δύο παίζουν βέλτιστα για να κερδίσουν;



Ο μπλε κερδίζει

Ο λευκός κερδίζει

Δεν μπορούμε να ξέρουμε

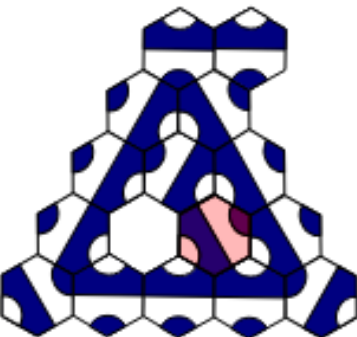
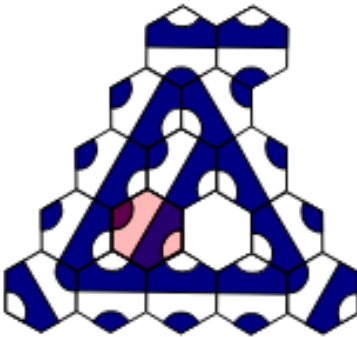
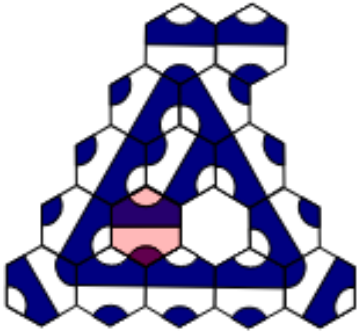
Ανάλογα ποιος έχει σειρά

## Απάντηση

Η επιλογή «Ο λευκός κερδίζει» είναι σωστή.

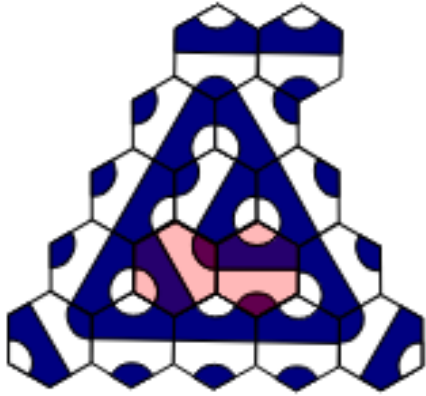
## Εξήγηση απάντησης

Έστω πως έχει σειρά ο λευκός, ο παίκτης μπορεί να ολοκληρώσει ένα λευκό κλειστό σχήμα με ένα μόνο κομμάτι χωρίς να κλείσει ένα μπλε σχήμα. Αυτό μπορεί να το κάνει με τρεις τρόπους:



Λόγω των κανόνων, ο παίκτης δεν χρειάζεται να τοποθετήσει το δεύτερο κομμάτι.

Τώρα έστω πως έχει σειρά ο μπλε, μπορεί να γεμίσει το κενό και να κλείσει το σχήμα και με τα δύο κομμάτια. Αλλά τότε θα έκλεινε επίσης ένα σχήμα για τον λευκό παίκτη, οπότε θα έχανε το παιχνίδι.



Αν ο μπλε παίκτης δεν κλείσει ένα σχήμα και το παιχνίδι συνεχιστεί, ο λευκός μπορεί να το κάνει στην επόμενη σειρά (βλ. παραπάνω). Ο μπλε δεν μπορεί απλά να βάλει ένα κομμάτι στο κενό επειδή τα δύο κομμάτια που τοποθετούνται πρέπει να ακουμπήσουν. Επίσης, ο μπλε δεν έχει τρόπο να κλείσει ένα σχήμα γύρω από τα υπόλοιπα κομμάτια που έχουν παιχτεί, οπότε ο λευκός μπορεί να κερδίσει το αργότερο στον επόμενο γύρο.

## Σύνδεση με την Υπολογιστική Σκέψη

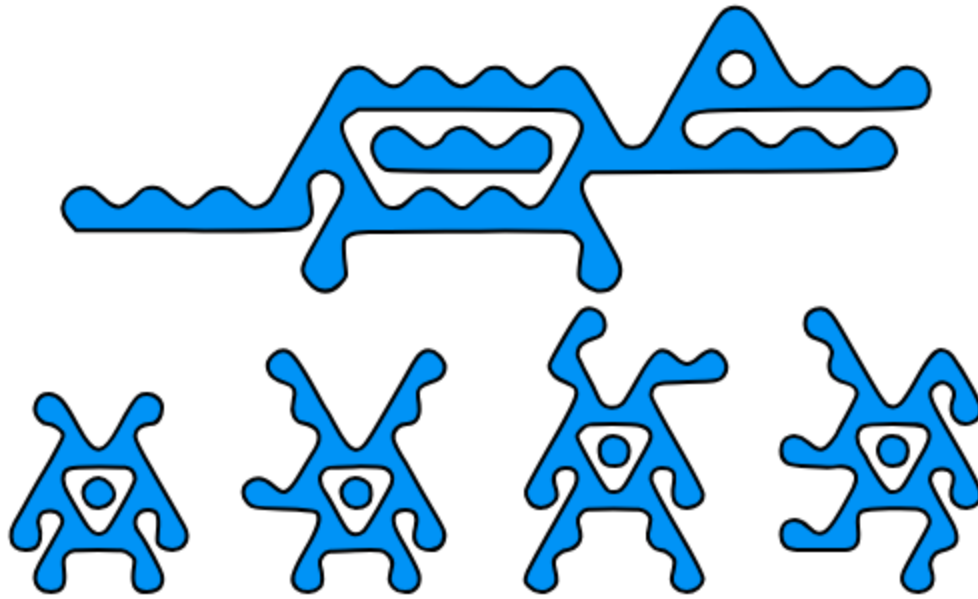
### Επιστήμη της πληροφορικής

Σε ένα παιχνίδι στρατηγικής για πολλούς παίκτες όπως αυτό (Palago) χρειάζεστε μια στρατηγική για να κερδίσετε ή να αποφύγετε την ήττα στα επόμενα βήματα. Είναι σημαντικό να βλέπετε τα επόμενα πιθανά βήματα και τα αποτελέσματά τους και να αποφασίζετε με βάση αυτά.

Για τα περισσότερα παιχνίδια στρατηγικής, αυτό μπορεί να διερευνηθεί συστηματικά χρησιμοποιώντας ένα δέντρο παιχνιδιών. Η ρίζα του δέντρου παιχνιδιών είναι η αρχική θέση και όλες οι πιθανές κινήσεις από κάθε θέση είναι τα κλαδιά. Έτσι δημιουργείται ένα δέντρο που αναπαριστά όλες τις πιθανές καταστάσεις του παιχνιδιού μέσα σε ένα τέτοιο παιχνίδι. Χτίζοντας ένα δέντρο παιχνιδιών, μπορείτε να βρείτε ένα μονοπάτι που μπορείτε να ακολουθήσετε για να κερδίσετε χωρίς να κερδίσει πρώτα ο αντίπαλός σας.

Ορισμένα παιχνίδια (σκάκι, Go και Palago) έχουν ένα τεράστιο ή ακόμα και απείρωσ μεγάλο δέντρο παιχνιδιών, οπότε πρέπει να εργαστείτε με ένα μερικό δέντρο παιχνιδιών και να αναλύσετε μόνο ένα μέρος ολόκληρου του δέντρου, για παράδειγμα μόνο τις επόμενες 5 κινήσεις. Εναλλακτικά, μπορείτε να αξιολογήσετε μια κατάσταση παιχνιδιού από έξω και να επιλέξετε μόνο εκείνες που υπόσχονται την επιτυχία.

Το Palago είναι ένα δημιουργικό παζλ τέχνης που σχεδιάστηκε από τον Cameron Browne. Δεν χρειάζεται να παίξετε το Palago μόνο για να κερδίσετε. Πολλοί παίκτες προσπαθούν να δημιουργήσουν έναν Παλαγονιανό (πολίτη της Παλαγονίας στον κόσμο του Palago) ή ζώα από τις πέτρες:



Χώρα προέλευσης θέματος

Ουγγαρία