

























Nakış Makinesi

Luna'nın evinde programlanabilir bir nakış makinesi var. Makine,  veya  şeklindeki dikişleri işleyebilir ve kumaşı 1 ilmek boyutunda hareket ettirebilir. Bu dikişlerin ikisini de aynı yere (herhangi bir sırayla) dikebiliriz, bir tane  elde edebiliriz. Makinenin programı, +, x ve → karakterlerinin bir dizisidir. + şekli  dikmek, x şekli  dikmek ve → kumaşı bir dikiş boyutunda hareket ettirmek anlamına gelir. Pedal basılı tutulurken makine girilen programı tekrarlar. Örneğin, makinede +→+x→x→ programına girersek, aşağıdakileri işleyecektir:                   .

Soru

Luna aşağıdaki nakışı dikebilmek için makineye hangi programı girmiştir?

A) x→x→x+→x→+x→x→x→

B) x→x→x+→x→x→

C) x→x→x+→x→+x→

D) +x→+x→x→+x→x→



Cevap Açıklaması

YANIT C

Doğru cevap, çıktıdaki deseni bularak bulunabilir.

Doğru cevap C'dir). Nakışa baktığımızda, yinelenen desen $\times \times * \times *$. İlk dikişin ardından kumaşın hareket ettirilmesi gerekir, bu nedenle program $x \rightarrow$ ile başlar. İkinci dikiş için de aynı işlem tekrarlanır. Sonra bir yıldız işlememiz gerekiyor - bu aynı yere x ve $+$ işleyerek yapılabilir (sıra önemli değil) ve ancak o zaman kumaş taşınır. Bunun için programımıza $x+ \rightarrow$ veya $+x \rightarrow$ ekleriz. Başka bir çarpı $x \rightarrow$ komutlarıyla ve bir yıldız da $+x \rightarrow$ veya $x+ \rightarrow$ komutlarıyla "yapılır". Yani tüm program $x \rightarrow x \rightarrow x+ \rightarrow x \rightarrow +x \rightarrow$ şeklindedir.

b) Seçenek A : $x \rightarrow x \rightarrow x+ \rightarrow x \rightarrow +x \rightarrow x \rightarrow x \rightarrow$ yanlıştır çünkü $x x * x * x * x x$ dizisini işlemektedir, bu Lana'nın sonuçta elde ettiği doğru bir desen değildir.

$\times \times * \times * \times * \times \times$

c) Seçenek B: $x \rightarrow x \rightarrow x+ \rightarrow x \rightarrow x \rightarrow$ yazılırsa, makine Lana'nın sonuçta elde ettiği doğru desen olmayan $x x * x x$ dizisini işleyecektir.

$\times \times * \times \times$

d) Seçenek D: $+x \rightarrow +x \rightarrow x \rightarrow +x \rightarrow x \rightarrow$ "+x→+x→" programının başlangıcında Lana'nın nakışında olmayan yan yana iki yıldız işleyecektir.

Enformatik Kavramı

Bu görev, algoritmaların ve örüntü tanımanın hesaplamalı düşünme kavramlarını gösterir. Örüntü tanıma, ya çözümdeki döngüler biçiminde ya da daha önce çözülmüş sorunlardan çözüm parçalarını yeniden kullanarak, çözümde yeniden kullanıma izin verecek olan sorundaki kalıpları bulma kavramıdır. Bu görevde, dikiş sırası bir desen oluşturur. Program, bir programlama dilindeki talimatların bir listesidir. Talimatları takip etmek bilgisayar bilimlerinde çok önemli bir kavramdır. Talimatların sırası çok önemlidir. Sırayı değiştirerek programın çıktısını değiştirebiliriz.

Farklı komut dizisi, farklı dikiş dizisi üretir - bu, farklı nakış anlamına gelir. Bu görevdeki "program" (kalıp), farklı yürütüldüğünde farklı sonuç verecek bir kalıptır, bu durum da pedala basıldığında gösterilir.

Anahtar Kelimeler ve Web Siteleri

Model, algoritma

<https://en.wikipedia.org/wiki/Pattern>

<https://en.wikipedia.org/wiki/Algorithm>

https://en.wikipedia.org/wiki/Pattern_recognition

Kelimeler

dikiş, nakış makinesi, nakış, program

Yazarlar, Katkı Sağlayanlar ve Editörler (Grafikler Dahil)

Yazar: Daniela Bezáková, daniela.bezakova@fmph.uniba.sk, Slovakia

Yorumcular : Victor Koleszar, Madhavan Mukund, Jifri Vaníček, Inggriani Liem, inge@informatika.org



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Kalp Şekli

Tuna'nın elinde, bir daire ve bir kare şekli bulunmaktadır.



Sonra bu şekilleri kullanarak aşağıdaki kalp şeklini yaratmıştır:



Tuna, bu şekilleri yapabilmek için aşağıdaki işlemleri kullandı.

- Döndür: Şekli herhangi bir yönde herhangi bir miktarda döndürün.
- Taşı: Şekli istediğiniz yere taşıyın.
- Çoğalt: Aynı konumda şeklin bir kopyasını oluşturun.

Soru

Tuna aşağıdaki işlemlerden hangisini kullanmış olabilir?

- A) Daireyi kopyala (çoğalt). Kareyi döndür. Daireyi hareket ettirin. Daireyi hareket ettirin.
- B) Kareyi kopyala (çoğalt). Kareyi döndür. Kareyi taşı. Daireyi hareket ettirin.
- C) Daireyi kopyala (çoğalt). Daireyi döndürün. Daireyi hareket ettirin. Kareyi taşı.
- D) Daireyi hareket ettirin. Daireyi hareket ettirin. Daireyi kopyalayın (çoğalt). Kareyi taşı.



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

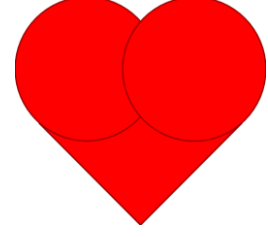
<https://creativecommons.org/licenses/by-sa/4.0/>

Cevap

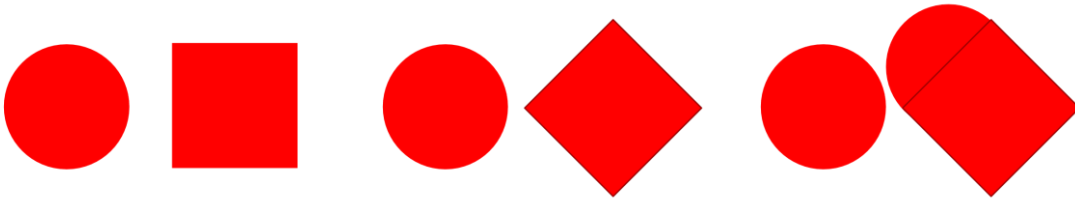
Açıklaması

YANIT: A

Karenin bir köşesi aşağı bakacak şekilde döndürülmesi gerekir. Bu sadece A ve B seçeneklerinde bulunmaktadır. Bu nedenle, doğru çözüm A veya B'dir. Ardından, bir kalp oluşturmak için iki daireye ihtiyacınız olduğunu görebilirsiniz. Bu sebeple B seçeneği doğru olamaz çünkü ikinci bir daire oluşturmaz. Bu, A'yı tek doğru seçenek olarak bırakır.



Aşağıdaki resimler, A seçeneğinin ilk üç işleminin her birinden sonraki şekilleri göstermektedir. Son dördüncü işlemde kalp oluşturulur. Bu, Tuna'nın A seçeneğini kullanmış olabileceğini doğrular.



Enformatik Kavramı

Bir grafik düzenleyici ile görüntüler oluşturduğunuzda, farklı işlemler gerçekleştirebilirsiniz. Bu görevin kalp görüntüsü için bir daireyi çoğaltın, bir kareyi döndürün ve daireleri hareket ettirin. Bir bilgisayarın işi yapmasını istiyorsanız, bazı bilgiler eklemeniz gerekir. Her daireyi nereye taşıyacağınızı ve kareyi ne kadar döndüreceğinizi belirtmeniz gerekir.

Cevap seçeneklerindeki işlem dizileri, ne yapılması gerektiğine dair eksik açıklamalardır. Bilgisayar bilimcileri, bunların komutların soyutlamaları olduğunu söylüyor. Ne yapılması gerektiği konusunda bir fikir veriyorlar ama bunu bir bilgisayarın yürütmesi için yeterince ayrıntılı olarak açıklamazlar. Bu yararlıdır çünkü ayrıntıları görmezden gelmek neyin önemli olduğuna odaklanmamızı sağlar.

Bilgisayar bilimcileri bir bilgisayar programı geliştirdiğinde, programın nasıl çalışacağı hakkında bir fikir edinmek için genellikle soyutlamalarla başlarlar ve daha sonra eksik ayrıntıları eklerler.

Anahtar Kelimeler ve Web Sayfaları

vektör grafikleri, şekil, algoritma, işlem

https://en.wikipedia.org/wiki/Graphics_software

Yazarlar ve Katkı Sağlayanlar

Michael Weigend (author), mw@creative-informatics.de, Germany

J.P. Pretti (Working Group 1), jpretti@uwaterloo.ca, Canada

Goran Sukovic (Working Group 1), goran.sukovic@gmail.com, Montenegro

Justina, Dauksaite (Working Group 1), jdauksaite@eljakim.nl, USA



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

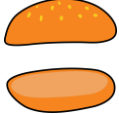






This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Hamburger Tarifi

Bilge Kunduz, hamburgerleri ařađıdaki kurallara gre yapıyor.

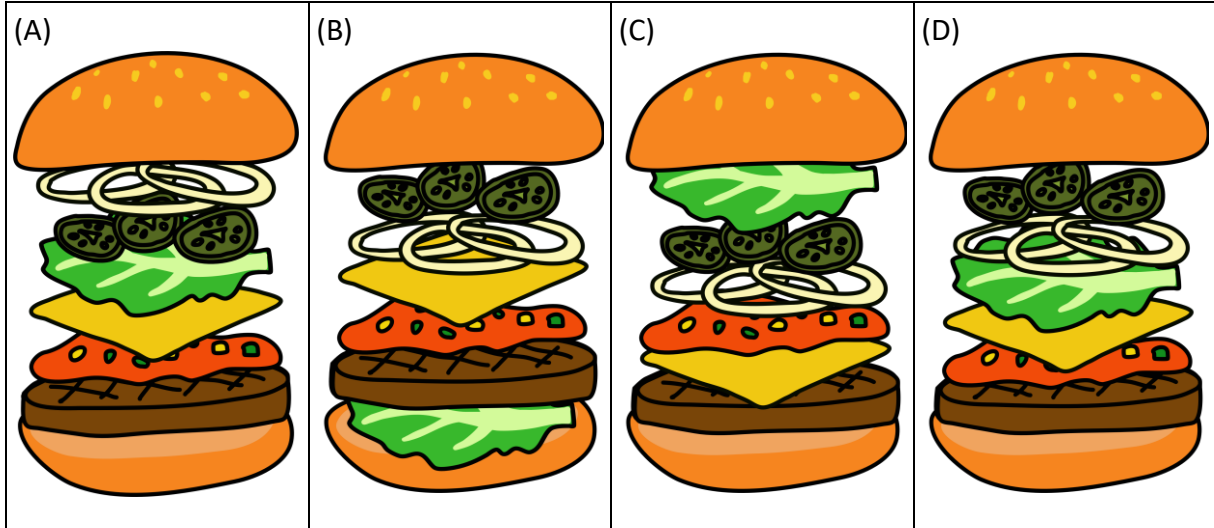
Hamburger malzemeleri:

Hamburger Ekmegi	Kfte	Sos	Turřu	Marul	Sođan	Peynir
						

1. Sos etin hemen zerinde olmalıdır.
2. Et ve peynir, turřu, marul ve sođanın altında olmalıdır.
3. Sođanlar ekmekler ile temas etmemelidir.
4. Tm malzemeler hamburger ekmeđi arasında olmalıdır.

Soru

Hangi hamburger kurallara gre dođru yapılmıřtır?



Cevap Açıklaması

Doğru cevap D'dir.

(A) Bu hamburger kural 1 ve 2'ye uyar. Ancak kural 3'e uymaz.

(B) Bu hamburger kural 1'e uyar, ancak marul etin ve peynirin altındadır, bu nedenle kural 2'ye uyulmamıştır.

(C) Peynir etle sos arasında olduğundan kural 1'e uymuyor.

(D) Tüm kurallara uyuyor, bu yüzden doğru cevap bu!

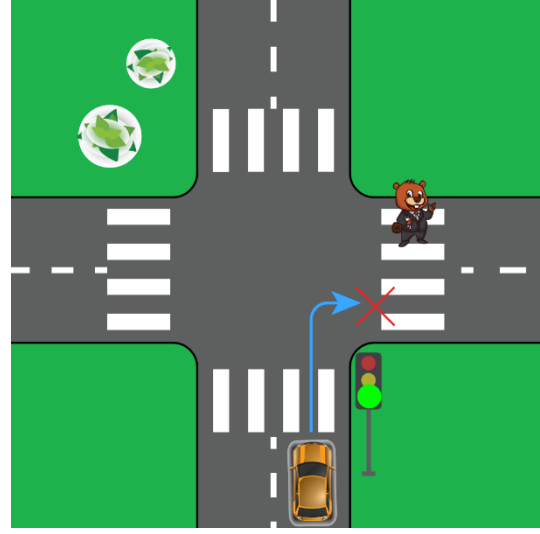
(Bu arada, her cevap kural 4'e uygundur.)

Enformatik Kavramı

Bilgisayar biliminde, bir çözümün verilen tüm kurallara uyup uymadığını bulmaya kısıtlama kontrolü denir. Yandaki şekilde kendi kendine giden bir araba sağa dönüp dönemeyeceğine karar vermek zorundadır. İki kural kullanır:

- Işık yeşil olmalıdır.
- Karşıdan yayalar geçmemelidir.

Belirli bir çözümün kısıtlamaları karşılayıp karşılamadığını kontrol etmek bir seçenek, böyle bir çözüm bulmak başka bir seçenektir. (Buna kısıtlama tatmin problemi denir.) Çoğu zaman bir çözüm bulmak, bir bilgisayar için bile kısıtlamaları kontrol etmekten çok daha zordur.



Anahtar Kelimeler ve Web Sayfası

Kısıt Memnuniyet Problemi

https://en.wikipedia.org/wiki/Constraint_satisfaction

Yazarlar, Katkıda Bulunanlar ve Editörler (Grafikler Dahil)

Byeonggyu Cho, cbg5946@gmail.com, South Korea

Seulki Kim, tmfrlska85@gmail.com, South Korea

Jihye Kim, anaki@korea.kr, South Korea

Hakin Kim, hakin711@gmail.com, South Korea

Ahto Truu (ahto.truu@ut.ee) Estonia, Ágnes Erdősne Németh (agi@microprof.hu) Hungary



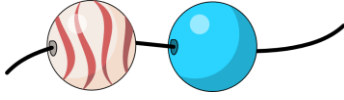
Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

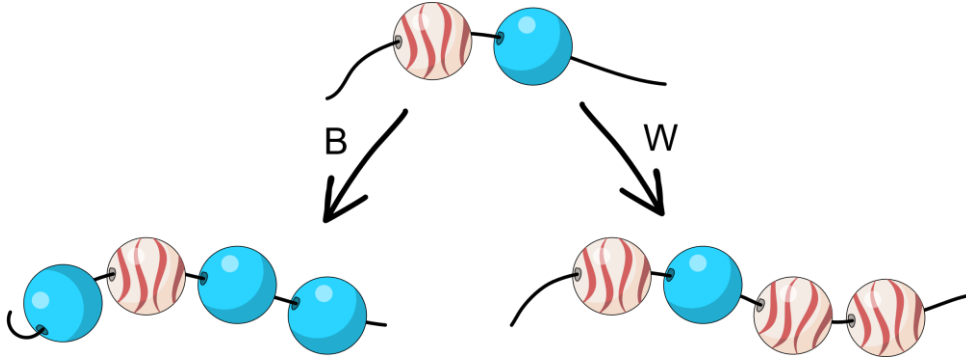
Denizci Kolyesi

Denizci kolyeleri dalgalı beyaz boncuklar ve düz mavi boncuklar kullanarak yapılır. Her denizci kolyesi, bir ipe bir dalgalı boncuk ve bir mavi boncuk gösterilen sırayla yerleştirilerek başlar:



Denizci kolyeleri aşağıdaki işlemlere uygun olarak yapılmalıdır.

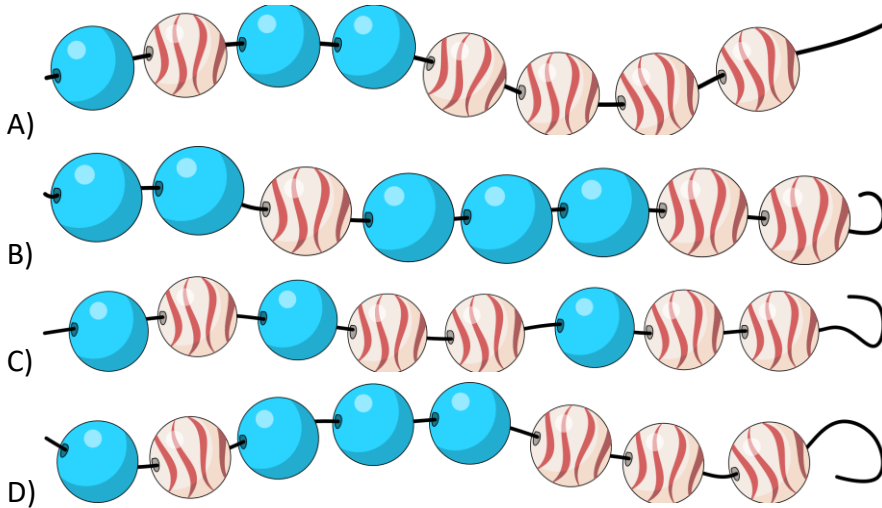
- dizinin her iki ucuna da mavi bir boncuk eklenmesi (eylem B)
- veya dizinin en sağ ucuna iki dalgalı boncuk ekleme (eylem W)



Bu işlemler, daha da uzun kolyeler oluşturmak için birden çok kez yapılabilir.

Soru

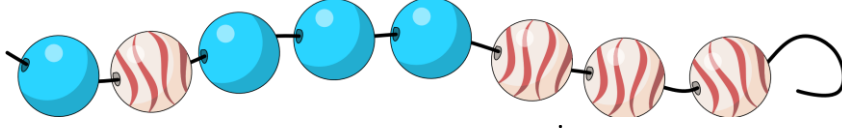
Aşağıdaki kolyelerden hangisi denizci kolyesi değildir?



Cevap Açıklaması

YANIT D

Doğru cevap D seçeneğidir.



Bu görevi çözebilmenin birçok yolu vardır. İşte üç olasılık:

1. Her bir kolyeyi, önce iki başlangıç boncuğu yerleştirerek ve ardından bir dizi B ve W eylemi gerçekleştirerek oluşturabilirsiniz. A kolyesi, ikinci ve üçüncü boncuklardan başlayarak ve ardından B-W-W eylemleri gerçekleştirerek yapılabilir. B kolyesi üçüncü ve dördüncü boncuklardan başlanarak ve ardından B-B W eylemleri gerçekleştirilerek yapılabilir. İkinci ve üçüncü boncuklardan başlanarak ve ardından W-B-W eylemleri gerçekleştirilerek C kolyesi yapılabilir. Ancak D kolyesine bakarsanız, başlangıç boncukları ikinci ve üçüncü boncuklar olmalıdır. B eylemini bir kez gerçekleştirebilirsiniz, ancak o zaman kolyenin geri kalanını oluşturacak hiçbir eylem yoktur.
2. Bu yaklaşım, kolye çok uzunsa ve birçok olası başlangıç boncuğuna sahip olsaydı iyi sonuç vermezdi. Bu durumda, bunun yerine, yalnızca iki başlangıç boncuğunuz kalana kadar B ve W hareketlerini tersine çevirerek boncukları tekrar tekrar çıkardığınız bir yapı sökücü yaklaşımı deneyebilirsiniz.
3. Üçüncü bir strateji, boncukların paritesi hakkında bir şeyler fark etmektir. Denizci kolye talimatlarına göre, her zaman tek sayıda düz mavi boncuk ve tek sayıda dalgalı beyaz boncuk olacaktır. Nedenini görebiliyor musun? D kolyesi, her iki tür boncuktan da çift sayıdadır ve bu nedenle bir denizci kolyesi olamaz.

Enformatik Kavramı

Bu görevde sadece kolye ipinin uçlarına boncuk ekleyebilirsiniz. Ortasına boncuk yerleştiremezsiniz. Benzer şekilde, kolyeyi ayırmak isterseniz, kolye ipinin uçlarındaki boncukları çıkarmanız gerekir. Ortadan bir boncuğu kolayca çıkaramazsınız. Ortadan değil, uçlardan kolayca ekleyip çıkarabileceğiniz bu yapı türü bilgisayar bilimlerinde kullanılıyor ve bir adı var – çift uçlu kuyruk veya deque. Deques, bir tarayıcının geçmişini depolamak, yazdırma işlerini planlamak ve ayrıca matematik ifadelerinin geçerliliğini doğrulamak için kullanılabilir. Eşleşen parantezlerin kontrolü, denizci kolyelerinin geçerliliğini nasıl doğruladığınıza oldukça benzer bir şekilde yapılabilir.

Anahtar Kelimeler ve Web Siteleri

https://en.wikipedia.org/wiki/Double-ended_queue

Kelimeler

Kolye, boncuk



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Yazarlar, Katkı Saęlayanlar, ve Editörler (Grafikler Dahil)

Yazar: Monika Tomcsányiová, monika.tomcsanyiova@fmph.uniba.sk, Slovakia

Grafikler: Karolína Miková, karolina.mikova@fmph.uniba.sk, Slovakia

Leo Barichello, barichello@gmail.com, Brazil

Sarah Chan, sarah.chan@uwaterloo.ca, Canada

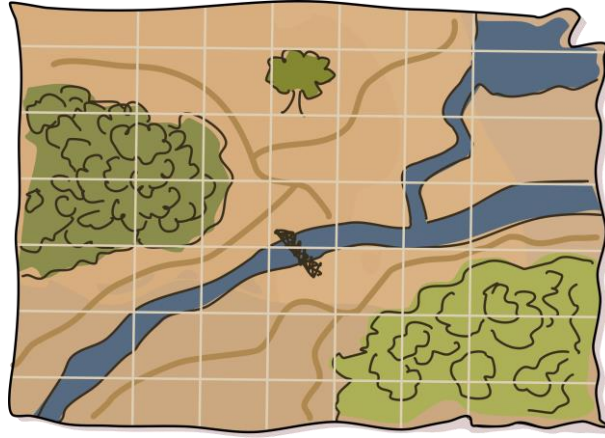
Susanne Datzko, susanne.datzko@informatik-biber.ch, Switzerland



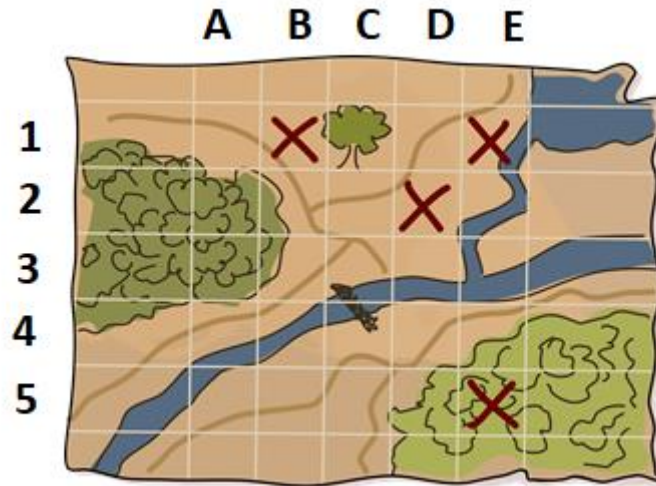
Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License
<https://creativecommons.org/licenses/by-sa/4.0/>

Kiler Haritası

Çağlar, yemekleri için iki adet saklama yeri buldu. Onları hatırlamak için bu haritadaki noktaları "X" ile işaretlemek istiyor. Ama rakibi Beril haritayı bulursa nereye bakacağını bilir.



Beril'i şaşırtmak isteyen Çağlar, her satırdaki ve her sütundaki toplam "X" sayısının çift olduğundan emin olarak, haritanın diğer karelerine rastgele bazı "X"ler ekler (Not: 0 da bir çift sayıdır). Sonra saklandığı yerleri gösteren iki "X"i siler. Ortaya çıkan harita bu:



Soru

Çağlar'ın iki saklanma yeri nerededir?

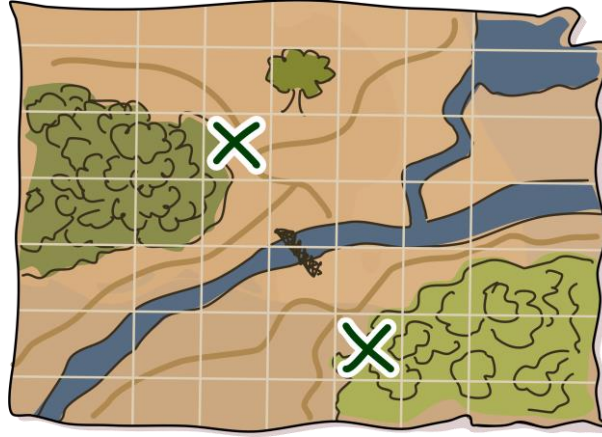
A) A3- E3 B) B2 - D5 C)A4 - D4 D) B3 - E



Cevap Açıklaması

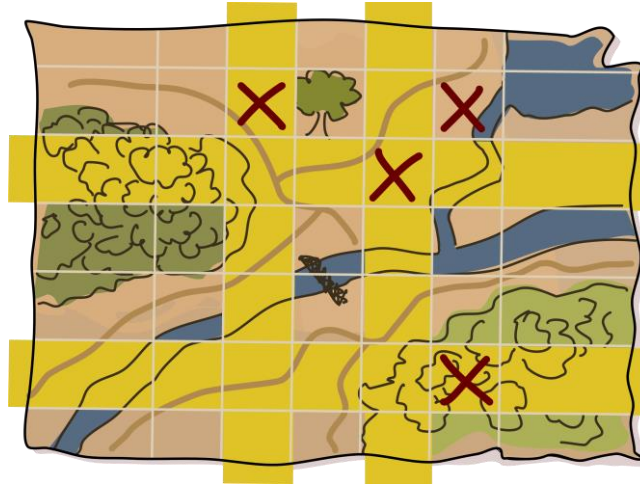
Yanıt B: B2 - D5

İşte iki saklanma yerinin konumu:



Onları bulmak için, son haritaya bakarız ve "X" sayısının eşit olmadığı iki satır ve iki sütun olduğunu görürüz: 2. ve 5. satırlar (alttan) ve 3. ve 5. sütunlar (soldan)

Bu görsel, sarı ile işaretlenmiş satırları ve sütunları gösterir:



Gerçek saklanma yerlerini gösteren iki silinmiş "X" onlarda olmalı! Bu yüzden, çift kısıtlamasını geri yüklemek için onları bu satırlara ve sütunlara geri eklemenin bir yolunu bulmalıyız.

Sarı sıra ve sütunların kesiştiği yerde 4 kare vardır. Üstteki kırmızı kesişme karesi zaten bir "X" ile işaretlenmiştir: Çağlar'ın "X"leri gerçek saklanma yerlerinden sildiğini bildiğimizden, bu bir saklanma yeri olamaz. Bu ikinci satırda çift sayıda "X" elde etmek için, onu sol üst sarı kesişme karesine eklemeliyiz: artık bir saklanma yeri biliyoruz.

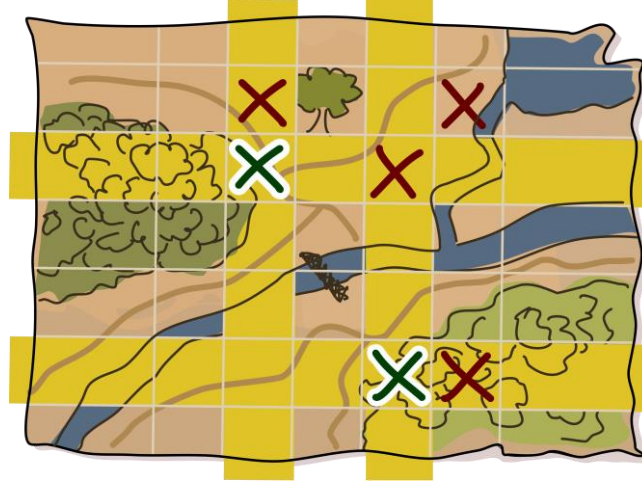
İkinci saklanma yeri şu şekilde belirlenebilir: birinciyi bulduktan sonra, ikinciyi sol alt kırmızı kesişme karesine yerleştiremeyiz, çünkü bu ikinci sütunda 3 "X" ile sonuçlanacaktır ve 3 bir çift sayı değildir. Soldaki tek seçenek, yukarıda gösterilen cevabı veren sağ alt kesişme karesidir.



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Bu, Çağlar'ın her satırda ve sütunda çift sayıda "X" ile "X"leri silmeden önceki haritayı gösterir:



Enformatik Kavramı

Çağlar, burada bilgisayar biliminde çok kullanılan eşlik biti kavramıyla ilgili bir hileyi, hata tespit ve hata düzeltme kodları olarak bilinen daha geniş bir teknikler setinin parçası olarak kullanıyor. Buradaki fikir, verileri bir dizi bit, 1'ler ve 0'lar olarak depolamamız veya aktarmamız gerektiğinde, aktarım veya depolama hatalarının oluşup oluşmadığını algılamamıza yardımcı olan ekstra bitler eklememizdir (tipik olarak, eğer bir bit ters çevrilmişse: örneğin, eğer 1 olarak iletildi ve hatalı olarak 0 olarak alındı).

Örneğin, basit bir hata tespit kodu kullanırsak, 1'lerin sayısının çift olması için bir eşlik biti eklenir. 0110101'e 0 eklenir ve 01101010 olur (1 bit sayısı çift tutulur). 2. bit çevrilmişse ve mesaj şimdi iletilmişse 00101010, o zaman bu alınan mesaj çift eşlik gereksinimini karşılamaz (3 bit 1'dir). İki bit hatalıysa, bu yöntemin bir sorun algılamayabileceğini unutmayın.

Anahtar Kelimeler ve Web Sayfaları

Parity Bit: https://en.wikipedia.org/wiki/Parity_bit

Hata düzeltme kodları: https://en.wikipedia.org/wiki/Error_detection_and_correction

Yazarlar, Katkı Sağlayanlar, ve Editörler (Grafikler Dahil)

Waël Almoman, wael.almoman@edu.ge.ch, Switzerland (original idea, graphics)

Jean-Philippe Pellet, jean-philippe.pellet@hepl.ch, Switzerland (rewrite, graphics modifications)

Susanne Datzko, susanne.datzko@informatik-biber.ch, Switzerland (new graphics)



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

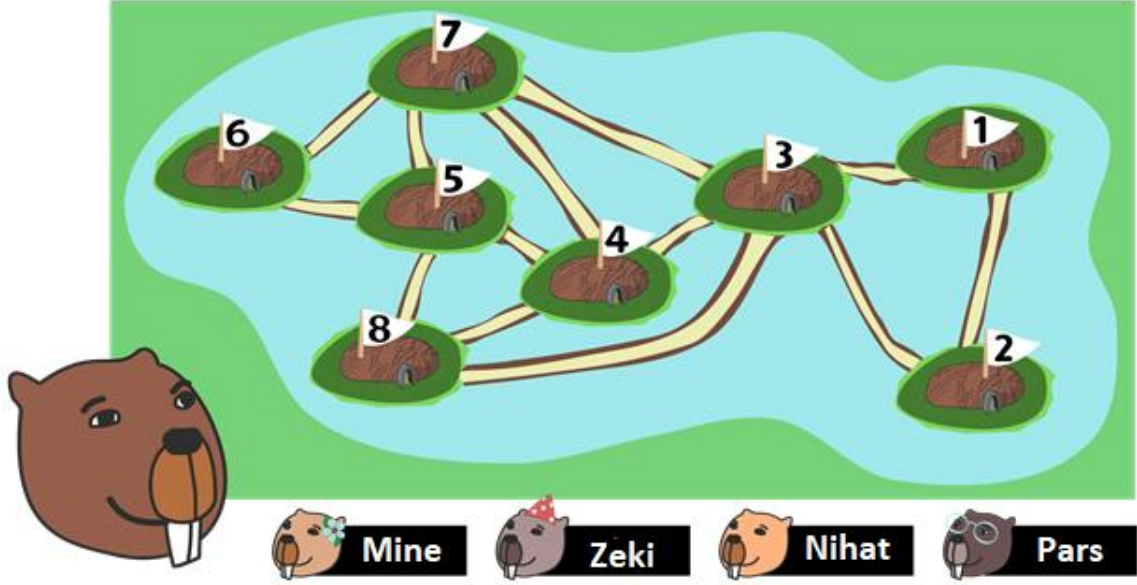
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Mine'nin Komşuları

Bulut, arkadaşı Mine'yi ziyaret etmek istiyor. Ama nerede yaşadığını bilmiyor. Neyse ki elinde bir harita ve bazı bilgiler var. Evlerini birbirine bağlayan bir yol varsa, iki kunduz komşudur.

- Aşağıdaki üç kunduzun her birinin: Mine, Zeki ve Pars'ın dört komşusu vardır.
- Zeki ve Pars, Nihat ile komşudur.
- Nihat'ın başka komşusu yoktur.



Soru

Mine'nin ev numarası nedir?

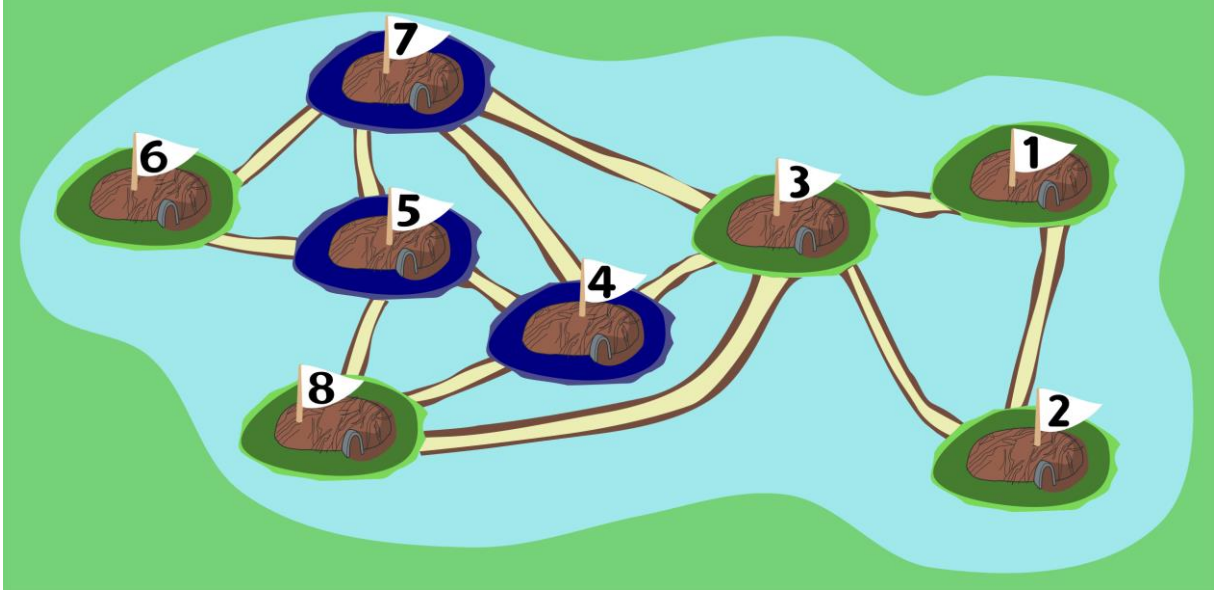
- A) 4 B) 5 C) 6 D) 7



Cevap Açıklaması

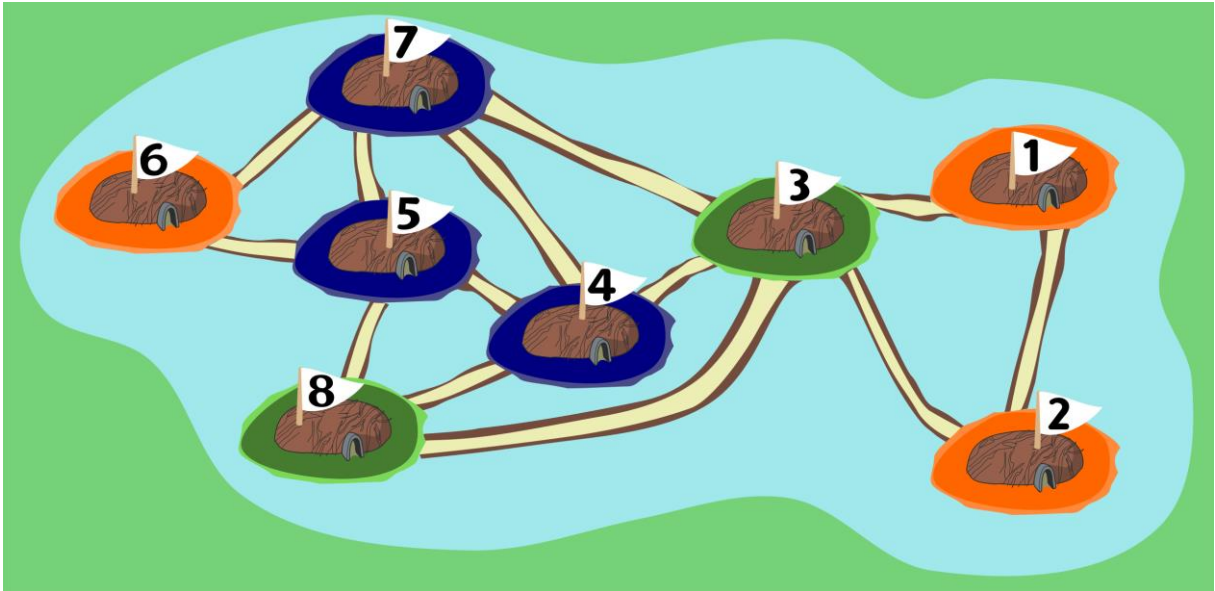
YANIT: A Cevap 4 numaradır.

Sorunu çözmek için her evden giden yollara odaklanmak gerekir. Önce dört yollu evleri tanımlamamız gerekiyor. Böyle üç ev var: 4, 5 ve 7.



Mine, Zeki ve Pars bu üç evden birinde yaşıyorlar ama Minen'in tam olarak nerede yaşadığını bulmamız gerekiyor.

Diğer iki bilgi Nihat'ın evini anlatıyor. Bundan sadece iki yol olduğu sonucunu çıkarabiliriz. Yani Nihat 1, 2 veya 6 numaralı evlerden birinde yaşıyor.



Nihat, Pars ve Zeki ile komşu olduğu için şu sonucu çıkarabiliriz:

Nihat 6 numaralı evde yaşıyor

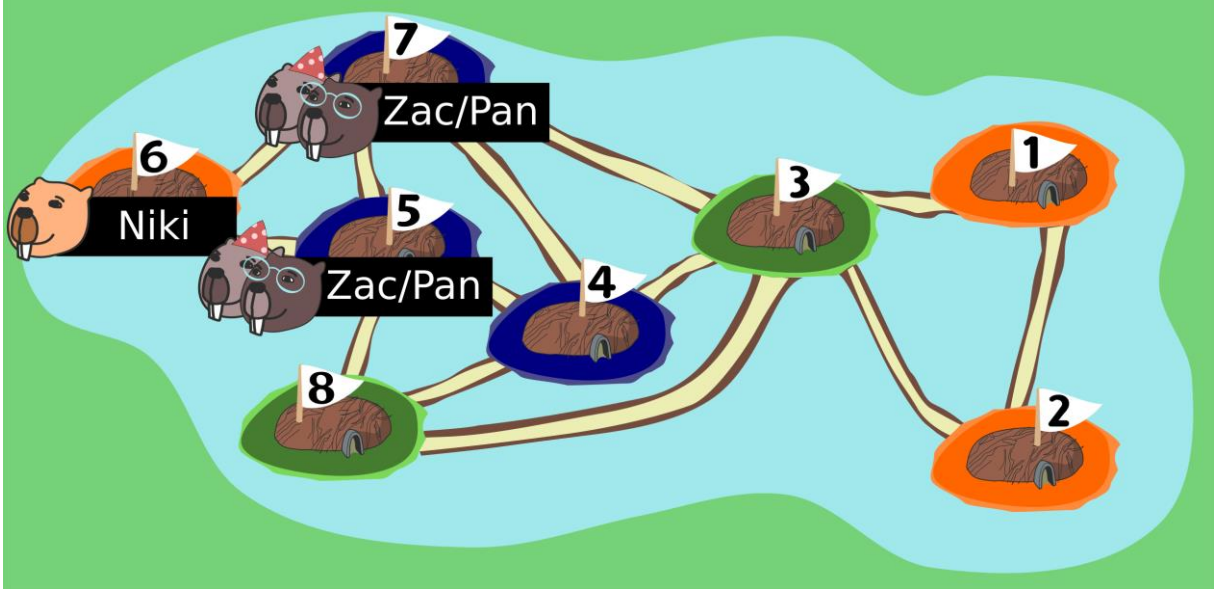
- Zeki ve Pars 5 ve 7 numaralarda yaşıyor (ya da tam tersi)



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

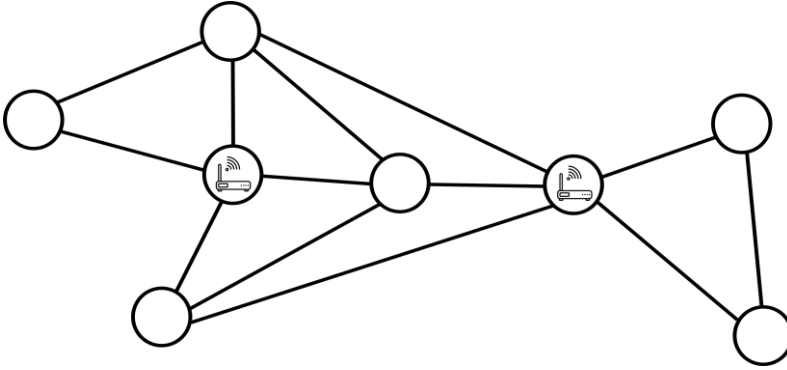
<https://creativecommons.org/licenses/by-sa/4.0/>



Böylece Mine'nın evi olabilecek dört yolu olan tek bir ev vardır. Ve 4 numaralı olan bu!

Enformatik Kavramı

Grafik teorisi, nesnelere arasındaki ikili ilişkileri modellemek için kullanılan grafiklerin incelenmesidir. Bir grafik, kenarlarla (bağlar veya çizgiler olarak da adlandırılır) bağlantılı bir dizi düğüm (köşe veya nokta olarak da adlandırılır) olarak görülebilir. Bu görevde evler düğümleri ve yollar kenarları temsil eder. Grafikler, bir binada bir yönlendirici için iyi bir nokta bulmak veya bir mahalledeki her bir evin güçlü bir wi-fi sinyaline sahip olduğundan emin olmak gibi ağ sorunlarını tanımlarken ve çözerken yardımcı olabilir.



Anahtar Kelimeler ve Web Sayfaları

https://en.wikipedia.org/wiki/Graph_theory

Grafik teorisi, kenarlar, düğümler.

Yazarlar, Katkı Sağlayanlar, ve Editörler (Grafikler Dahil)

Thomas Ioannou, ioannouthomas@gmail.com, Cyprus

Marielle Léonard, marielleleonard59@gmail.com, France

Marta J. Burzanska, quintria@mat.umk.pl, Poland



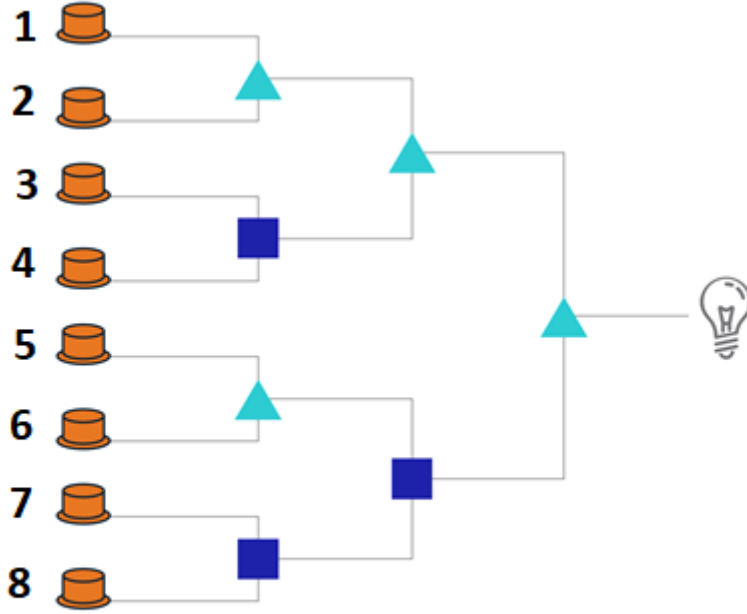
Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Mantık Kapıları

Kunduz Seda ve arkadaşları bir deney yapıyorlar. Deneyde Seda ve arkadaşları, 8 düğmeyi kontrol ederek kabloya bir sinyal gönderecekler. Bu kablolar bazı üçgen veya kare kutulardan geçerek bir ampulün yanmasını sağlar.



Gelen her iki kablo da sinyal gönderirse, üçgen kutu bir sinyal gönderir.
Gelen kablolardan yalnızca biri bir sinyal gönderirse, kare kutu bir sinyal gönderir.

Soru

Seda ve arkadaşları sonunda ampulü yakmak için hangi düğmelere basmalıdır?

- A) 1-2-4-5-6
- B) 1-2-3-4-5
- C) 1-3-4-5-6
- D) 1-3-5-7-8

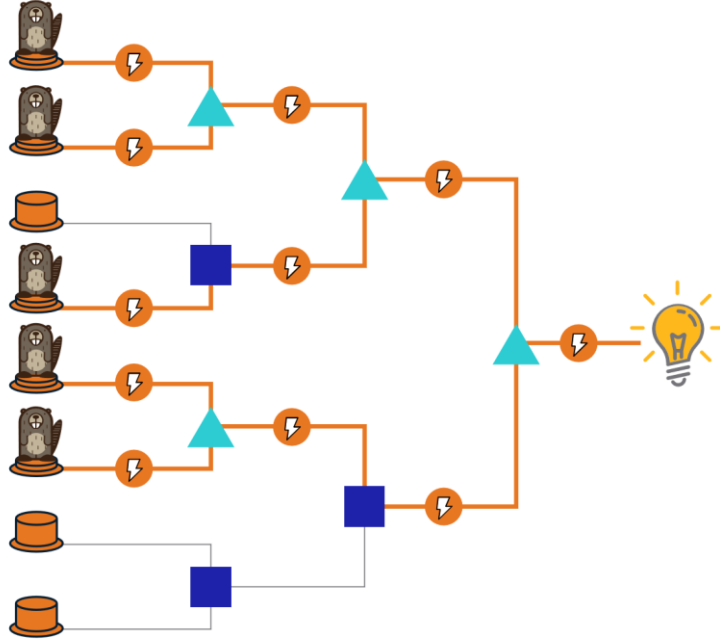
Yanıt : A



Doğru yanıt

Yanıt: A

Sonunda ampulü açacak 8 düğmenin 16 olası kombinasyonu vardır. Bu çözümlerden biri aşağıda gösterilmiştir.



Teller açıksa turuncu renkle vurgulanmış şekilde ışığı açan düğmeye basmanın ekran görüntüsü.

Sorunun Çözümü

Bu soruya iyi bir yaklaşım geriye doğru çalışmaktır. Uçtaki ampul, bir üçgenden gelen 4. sütundaki tele bağlanmıştır. Bu telin açık olması için üçgene giden iki telin de açık olması gerektiğini biliyoruz.

Bu teller bir üçgen ve bir kareye bağlanır.

Üçgenin açık olması gerektiğini biliyoruz, bu nedenle ona bağlanan kabloların her ikisi de açık olmalıdır.

Karenin açık olması gerektiğini biliyoruz, bu nedenle ona bağlanan tellerden biri açık, diğeri kapalı olmalıdır.

Sütun 1 ve 2 için, üst yarıdaki ve alt yarıdaki tellere ayrı ayrı bakacağız.

Üst yarı için: Sütun 2'deki her iki kablo da açık olmalıdır. Bu nedenle, 1. sütundaki en üstteki 2 düğmeye basılmalı ve en alttaki 2 düğmenin düğmelerinden tam olarak birine basılmalıdır.

Alt yarı için: Kare açık konumdayken, içine giden tellerden tam olarak biri açık, diğeri kapalı olmalıdır. Bu nedenle, ya ona giden üçgen açık ve kare kapalı olmalıdır ya da tam tersi. Bunu yapmanın birden çok yolu vardır – yukarıda gösterilen çözüm olası bir yaklaşımı göstermektedir. Burada üçgen açıktır ve bu nedenle üstteki iki düğmeye basılmalıdır, yani kare kapalı olmalıdır ve alttaki iki düğmenin her ikisine de basılır veya basılmaz (bu durumda ikisine de basılmaz).

Sorudaki Enformatik Kavramı

Bu soruda, teller açık veya kapalı olabilir. Bilgisayar bilimcileri, yalnızca iki farklı durumda olabilen bu tür şeylere Boolean verileri derler. Bunun diğer örnekleri, ikili sayılar (0'lar ve 1'ler kullanılarak) ve bir şeyin doğru veya yanlış olup olmadığına bakmaktır.



Boole verileri, bilgisayarların nasıl çalıştığı konusunda çok önemlidir. Bilgisayarlar, transistör adı verilen milyarlarca küçük anahtardan oluşur. Bu transistörler kapalı veya açık olabilir ve bir bilgisayarın yapabileceği her şey, bu iki durum arasında değişen transistör kombinasyonlarından ibarettir.

Bilgisayarlar ayrıca bu soruda gördüğümüz bir diğer önemli bileşeni kullanır. Sadece gelen tellerin ikisi de açıkken açılan üçgenlere AND kapısı, sadece bir tane gelen tel açıkken açılan karelere ise XOR kapısı denir. Bunlar, kendisine bağlı diğer tellerin açık veya kapalı olmasına bağlı olarak bir kabloyu açan birkaç kapı türünden sadece ikisidir – bu kapılar Boolean Logic örnekleridir.

Boolean Logic'i görebileceğimiz bir başka yer de bilgisayar programlarıdır. Bazen bir programın daha önce bir şeyin (veya bazen birçok şeyin) olup olmadığına bağlı olarak bir sonraki adımda ne yapılacağına "karar vermesi" gerekebilir. Bilgisayar programcılarının bunu programlarına koymasının iki yaygın yolu, IF deyimlerini veya AND deyimlerini kullanmaktır.

Anahtar Kelimeler ve İlgili Web Siteleri

Boolean, mantık, kapılar, veri, IF, AND, XOR, transistör.

<https://www.khanacademy.org/computing/computers-and-internet/xcae6f4a7ff015e7d:computers/xcae6f4a7ff015e7d:logic-gates-and-circuits/a/logic-gates>

https://en.wikipedia.org/wiki/Logic_gate

https://en.wikipedia.org/wiki/Boolean_data_type

Yazarlar ve Katkı Sağlayanlar

Adam Grodeck (author), adam.grodeck@csiro.au, Avustralya

Susannah Quidilla: contributor, susannah.quidilla@csiro.au, Avustralya



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Somun ve Cıvata

Kunduz İnşaat fabrikasında... Bilge Kunduz, somun ve cıvata montaj hattında çalışıyor.



Görev tanımı aşağıdaki gibidir:

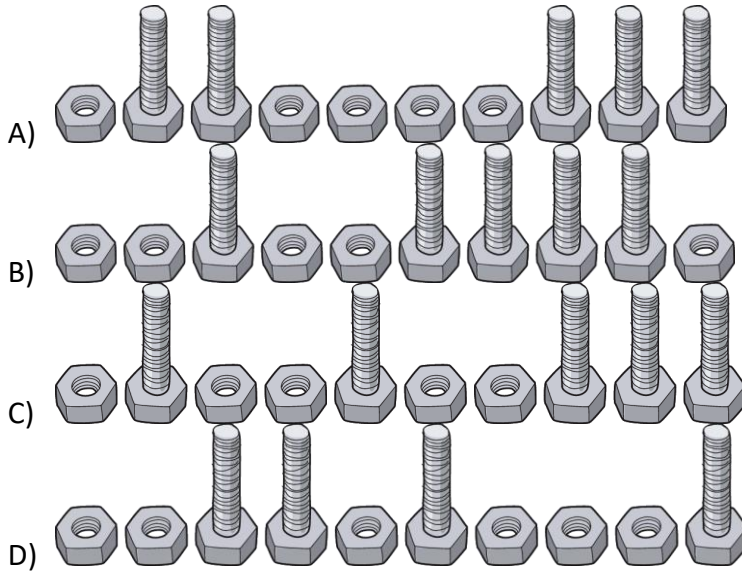
- Bilge Kunduz, bir dizi somun ve cıvata içeren uzun bir konveyör bandının bir ucunda duruyor.
- Bilge Kunduz'un işi, bir somun veya bir cıvata gibi her bir elemanı konveyör bandından çıkarmaktır.
- Bilge Kunduz, konveyör banttın bir somun alırsa, yanındaki kovaya koyar.
- Bilge Kunduz, konveyör banttın bir cıvata alırsa yanındaki kovadan bir somun alır, somun ve cıvatayı birbirine bağlar ve monte edilen parçayı büyük kutunun üzerine yerleştirir.

Ancak, Bilge Kunduz için işler iki farklı şekilde ters gidebilir:

1. Bilge Kunduz, konveyör banttın bir cıvata alırsa ve kovada takılacak somun yoksa.
2. Konveyör bandında artık somun veya cıvata yoksa ve kovada hala somunlar varsa.

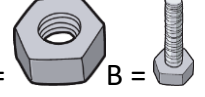
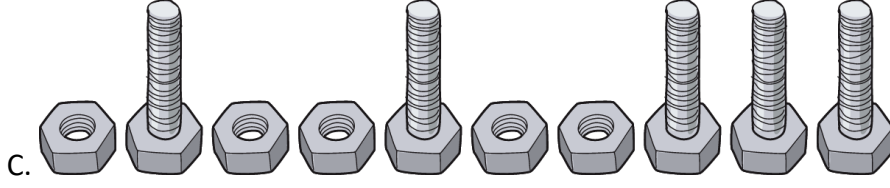
Soru

Hangi somun ve cıvata dizisi soldan sağa doğru işlendiğinde Bilge Kunduz için işlerin ters gitmesine neden olmaz?



AÇIKLAMA

YANIT



Kova (bucket) ve konveyör bandının durumunu soldan sağa takip edebiliriz: N =

Bucket	Conveyor Belt
empty	N B N N B N N B B B
N	B N N B N N B B B
empty	N N B N N B B B
N	N B N N B B B
N N	B N N B B B
N	N N B B B
N N	N B B B
N N N	B B B
N N	B B
N	B
empty	empty

Diğer cevaplara bakıldığında:

- A. N B B'den sonra yanlış gidecek, çünkü ikinci B ile karşılaşıldığında kovada somun olmayacak.
- B. N N B N N B B B B'den sonra yanlış gidecek, çünkü beşinci B ile karşılaşıldığında kovada somun olmayacak: Dikkat edin, bu B'den önce sadece 4 N var.
- D. 6 N'ler ve 4 B'ler olduğundan kovada iki somun olacağından tüm dizi işlemlerden sonra yanlış gidecektir.

Sorudaki Enformatik Kavramı

Bu görev, aşağı itilen otomatların (PDA) kullanımını vurgular. Bir PDA, mevcut duruma dayanan, ancak aynı zamanda yığın şeklinde sınırsız miktarda belleğe sahip bir algoritmayı tanımlamanın bir yoludur. Bu görevde, devlet ya bir somuna sahiptir ya da konveyör bandında bir civataya sahiptir ve yığın, somunları tutan kovadır.

Bağlamdan bağımsız dilleri tanımak veya ayrıştırmak için bir PDA kullanılabilir. Bir dili tanımak veya ayrıştırmak, belirli bir sembol dizisinin dile ait olup olmadığını belirlemek anlamına gelir. Bu durumda, somunları ve civataları, N=(ve B=) olmak üzere, dengeli parantezlerin temsili olarak düşünebiliriz. Yani, dengeli parantezler, aritmetik ifadelerde geçerli parantez düzenlemeleridir. Dengeli olmayan bir parantez dizisinin örnekleri (((() veya



()(). Birçok programlama dili, aritmetik ifadelerin yanı sıra iç içe kapsamları belirtmek için parantezleri kullandığından, derleyicilerde dengeli parantezleri algılamak önemlidir.

Anahtar Kelimeler

Push-down automata (PDA) or push-down automation:

https://en.wikipedia.org/wiki/Pushdown_automaton

Automata: https://en.wikipedia.org/wiki/Automata_theory

parsing, context-free languages, context-free grammars, balanced parentheses, stack

Yazarlar ve Katkı Sağlayanlar

Troy Vasiga, troy.vasiga@uwaterloo.ca, Canada

Graphics, ITW 2022 WG: Vaidotas Kincius, vaidotas.kincius@gmail.com, Lithuania

Editor, ITW 2022 WG O4: Taina Lehtimäki, taina@cs.nuim.ie, Ireland

Editor, ITW 2022 WG O4: Linda Björk Bergsveinsdóttir, linda@sky.is, Iceland



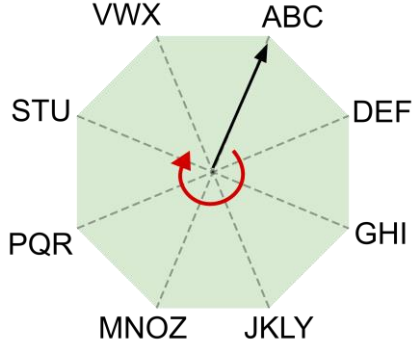
Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Şifreleme

Mesaj şifrelemek için kullandığımız sekizgen şeklinde bir aracımız var. Sekizgenin her köşesinde üç veya dört harften oluşan harf grupları bulunuyor. Saat yönünde dönebilen bir ok yardımı ile harf gruplarını gösterip mesajları şifreleyebiliyoruz.



Her yeni mesaja başlarken ok her zaman ABC harflerini gösterir.

Mesajın her harfini şifreliyoruz, böylece:

- İlk sayı, okun mevcut dönüşünden itibaren sekizgenin kaç köşesini geçtiğimizi gösterir.
- İkinci sayı, okun gösterdiği harf grubundaki şifreli harfin konumu anlamına gelir.

- Şifrelenmiş harfler "-" ile ayrılır.

Örneğin, TREE mesajı 62-73-42-02 dizisiyle şifrelenmiştir.

Soru

WATER mesajını nasıl şifreleriz?

- A) 72-11-26-32-53
- B) 62-11-62-22-43
- C) 62-11-26-22-53
- D) 72-11-62-32-43



Cevap Açıklaması

Doğru cevap D) 72-11-62-32-43

WATER mesajı için kademeli olarak şifreli bir metin oluşturarak doğru cevabı alıyoruz.

Şifrelenmiş metin 5 kod içerecektir:

- Şifrelemenin başlangıcında ok, ABC harf grubunu gösterir. W harfi, okun 7 köşe döndürüldükten sonra gösterileceği harf grubu içinde yer alır. W harfi VWX harf grubundaki ikinci harftir, yani ilk kod 72'dir.
- Şifreli mesajın ikinci harfi A'dır. Bu harf, VWX harf grubundaki mevcut konumundan bir kez döndürüldükten sonra okun göstereceği ABC harf grubundadır. A birinci konumdadır, dolayısıyla ikinci kod 11'dir.
- 6 köşeyi döndürerek T içeren harf grubuna ulaşıyoruz ve T harfi ikinci harf, yani üçüncü kod 62.
- E harfi, ok 3 köşe döndürüldükten sonra okun göstereceği harf grubundadır ve E, gruptaki ikinci harftir, yani kod 32'dir.
- 4 köşe döndürüldükten sonra ulaşılan harf grubunda son harf R, üçüncü harf R yani son kod 43'tür.

Şifreli metnin tamamı 72-11-62-32-53

Enformatik Kavramı

Verileri yetkisiz kişilerden korumanın bir yöntemi gizli şifrelemedir. Kriptografi 3500 yıl önce başladı. En basit şifreleme yöntemlerinden biri, her harfi farklı bir harfle değiştirmektir. Problemimizde, okun dönüşünden dolayı, kelimedeki bu harften önce gelen harfe bağlı olarak bir harf için farklı bir şifreli mesaj veren bir şifreleme yöntemi tasarlanmıştır. Bu nedenle, her kod okun daha önce hangi harf üçlüsüne işaret ettiğine ve dolayısıyla, şifrelenen kelimenin ilk harfi için okun harf üçlüsüne işaret etmek üzere ne kadar döndürülmesi gerektiğine bağlıdır. Kelimenin geri kalan harfleri aynı kodlara sahip olsa da, bu şekilde tasarlanan şifrede bir miktar değişkenlik vardır ve bu nedenle tespit edilmesi daha zor olabilir. Ancak bu şekilde tasarlanan şifre o kadar basit ki, bir kişinin kolayca hatırlaması sorun değildir.

Bu problemde, her köşedeki harfleri değiştirirsek veya farklı üçlülere koyarsak, başka şifreleme yöntemleri elde ederiz. Böylece kodları kırmaya çalışan kriptanalitikler, şifreleme düzenimizi çözmede sorun yaşayabilir.

Anahtar Kelimeler ve Web Sayfaları

Kriptografi, şifre, veri güvenliği, şifreli metin

<https://en.wikipedia.org/wiki/Cryptography>

Yazarlar, Katkı Sağlayanlar, ve Editörler (Grafikler Dahil)

Author: Monika Tomcsányiová, monika.tomcsanyiova@fmph.uniba.sk, Slovakia

Editor: Liam Baumann, lbaumann@outlook.at, Austria



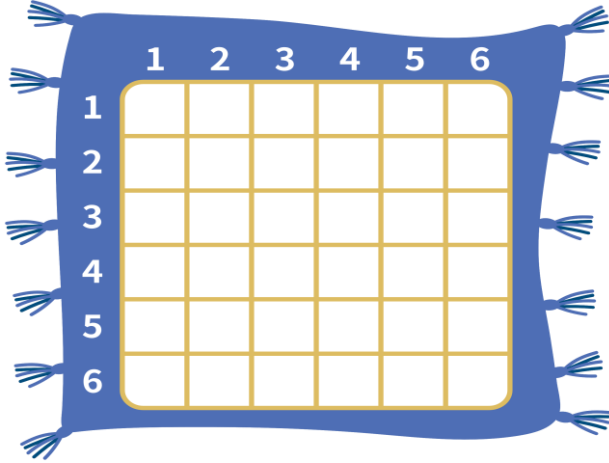
Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

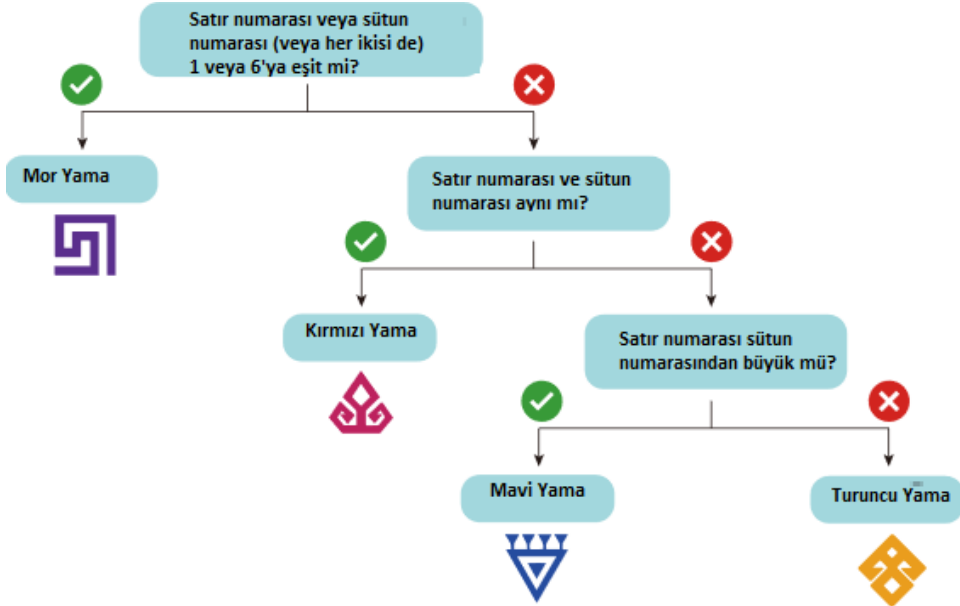
<https://creativecommons.org/licenses/by-sa/4.0/>

Kilim Dokuma

Hale bir Türk dokuma sanatçısıdır. 6 sıra ve 6 sütundan oluşan kare bir kilim dokur.



Hale, aşağıdaki soruları kullanarak halının içindeki her kareye bir sembol yerleştirir:



Soru

Bu yöntemi kullanarak ortaya çıkan halı aşağıdakilerden hangisidir?

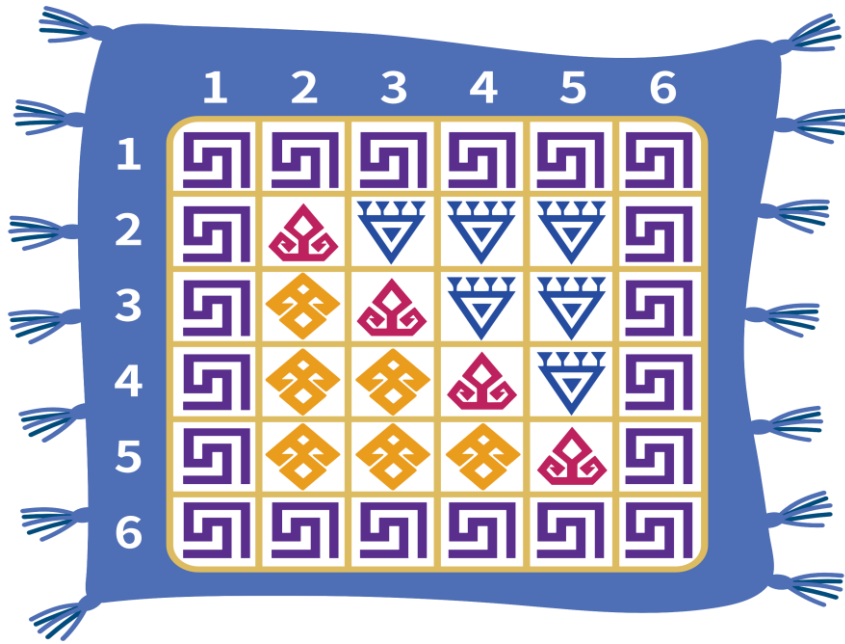
A)



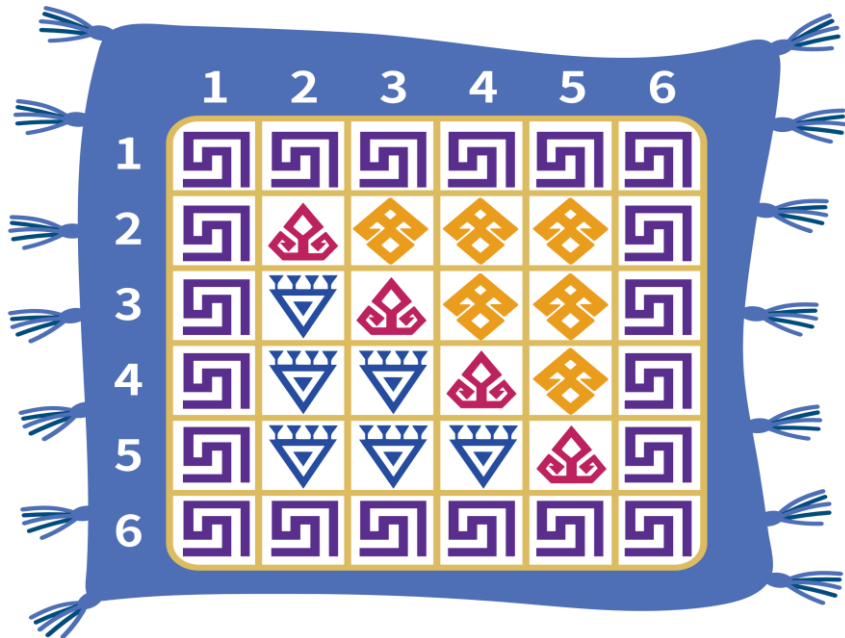
Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

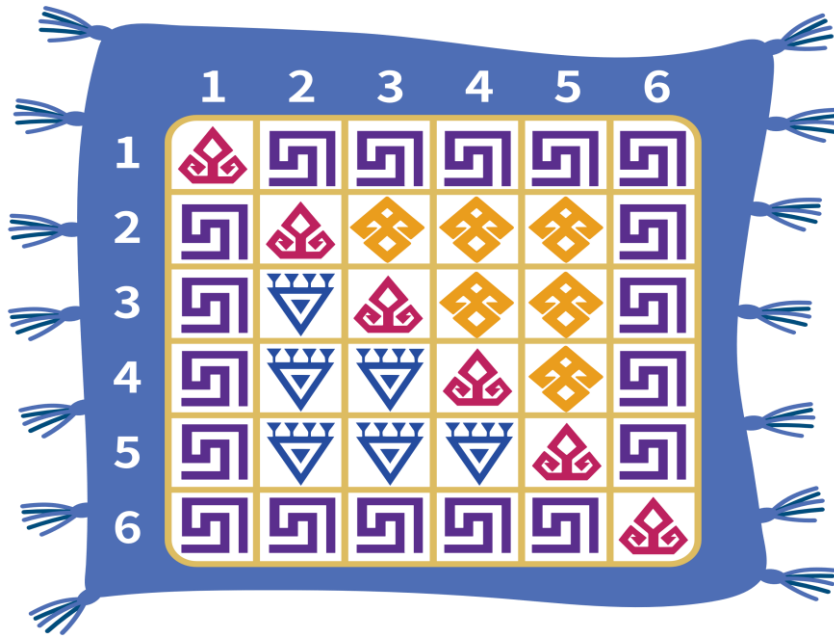


B)

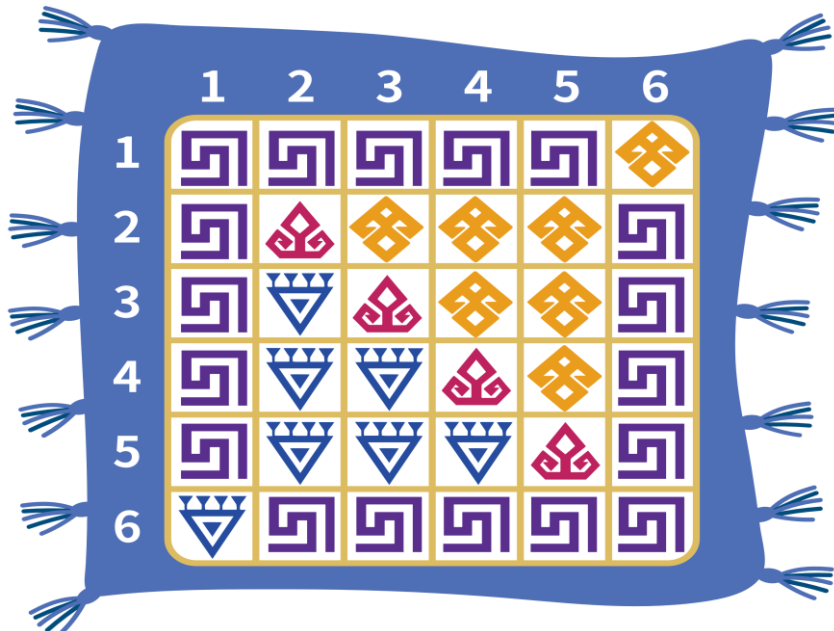


C)





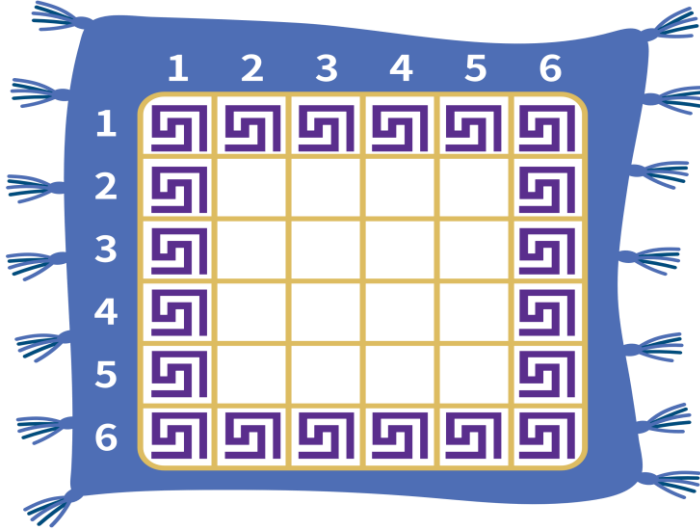
D)



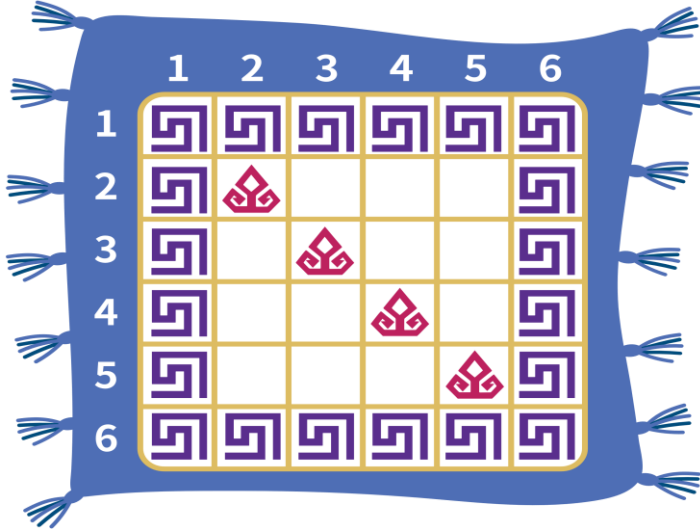
Cevap Açıklaması

Doğru Cevap: B

Hale'nin sorduğu ilk soru, 1. ve 6. sıraların yanı sıra 1. ve 6. sütunların da mor şekillere sahip olması gerektiğini gösteriyor. Bu, aşağıdaki sembol düzenlemesini verir:

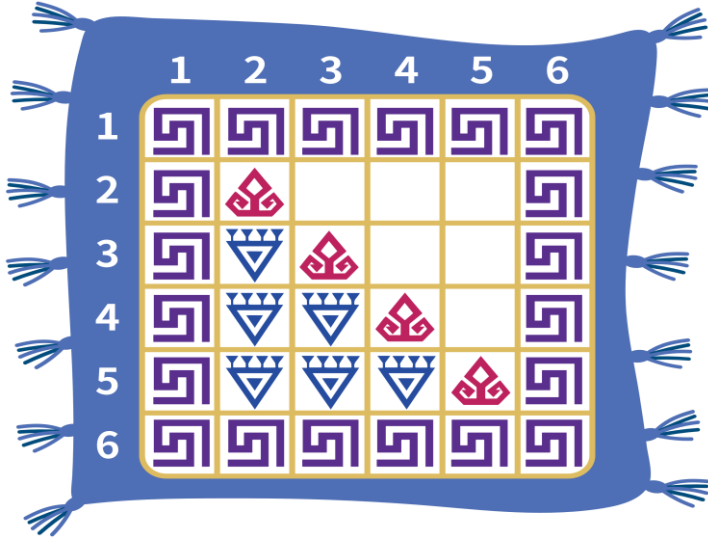


İkinci soru, halının köşegenindeki tüm karelerin kırmızı bir sembole sahip olmasına yol açar, çünkü köşegende satır ve sütun numaraları aynıdır:



Bir sonraki soru katmanında, köşegenin solundaki tüm karelerde satır numarasının sütun numarasından daha büyük olduğunu buluyoruz. Bu yerlere mavi bir sembol yerleştirilir:





Son olarak, köşegenin sağındaki kalan karelerin tümü, sütun numaralarından daha büyük OLMAYAN satır numaralarına sahiptir, bu nedenle, B seçeneğinde görülen yanıtı veren sarı sembollerle doldurulurlar.

Enformatik Kavramı

Matematik ve bilgisayar bilimlerinde bir algoritma, tipik olarak belirli bir problem sınıfını çözmek veya bir hesaplama yapmak için kullanılan, iyi tanımlanmış talimatların sonlu bir dizisidir. Algoritmalar, hesaplamalar yapmak ve veri işlemek için spesifikasyonlar olarak kullanılır.

Bir karar ağacı, IF/ELSE ifadeleri gibi koşullu ifadelerden oluşan bir algoritmanın yapısını kolayca görüntülemenin bir yoludur. Bu, karar ağacının her katmanının bir soru veya ifadeden oluştuğu anlamına gelir - soru/ifade doğruysa, algoritma ağacın bir dalını takip eder, aksi takdirde algoritma ağacın diğer dalını takip eder. Bu, algoritmanın sonucuna ulaşılan kadar farklı dallarda devam eder.

Algoritmalar, yapay zekayı kullanarak otomatik kesintiler yapabilir (otomatik akıl yürütme olarak adlandırılır) ve kodu çeşitli yollardan yönlendirmek için matematiksel ve mantıksal testler kullanabilir (otomatik karar verme olarak adlandırılır).

Anahtar Kelimeler ve Web Sayfaları

<https://en.wikipedia.org/wiki/Algorithm>
https://en.wikipedia.org/wiki/Decision_tree

Yazarlar, Katkı Sağlayanlar, ve Editöler (Grafikler Dahil)

Yazar: Yasemin Gulbahar, ysmnglbhr@gmail.com, Turkey

Katkı Sağlayanlar, Grafik Editörü: Gulgun Afacan, gulgunafacan@gmail.com, Turkey

Editör: Adam Grodeck, adam.grodeck@csiro.au, Australia

Grafik: Yeh Yi Shan, ishabecat@gmail.com, Taiwan.



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Listeler

3, 5, 2, 4, 1 sayılarının bir listesini aşağıdaki gibi görsel olarak temsil edebiliriz. (Sütunların üstündeki sayılar listedeki konumları gösterir.)

	1	2	3	4	5
X	3	5	2	4	1

2. konumdaki sayıyı tanımlamak için (X 2) yazıyoruz. Yani (X 2) 5'tir. Benzer şekilde (X 5) de 1'dir.

Konumlar dolaylı olarak belirtilebilir. Örneğin (X (X 3)) 5'tir çünkü (X 3) 2'dir, yani (X (X 3)) = (X 2) = 5.

İşte A, B ve C için üç liste.

A

3	2	4	1	5
---	---	---	---	---

B

5	4	1	3	2
---	---	---	---	---

C

2	5	4	3	1
---	---	---	---	---

Soru

(A (B (C 3))) ile açıklanan sayı nedir?

A) 2 B) 3 C) 4 D) 5



Cevap

Yanıt: C=4

Cevap Açıklaması

$(C\ 3) = 4$, yani $(B\ (C\ 3)) = (B\ 4) = 3$, yani $(A\ (B\ (C\ 3))) = (A\ (B\ 4)) = (A\ 3) = 4$

Enformatik Kavramı

Veri yapıları programlama için gereklidir. Özellikle tüm bir veri listesini tutabilenler yararlıdır. Bilgisayar bilimi ve programlamadaki diğer birçok varlık gibi veri yapıları birbirine bağlanabilir, böylece bir listenin bir öğesi başka bir listedeki bir konumu tanımlayabilir. Konumları tanımlamanın bu dolaylı yolu güçlü bir kavramdır. İlk başta kafa karıştırıcı görünebilir, ancak değerleri adım adım hesaplırsanız zor değildir.

Anahtar Kelimeler ve Web Sayfaları

Veri türleri, listeler, diziler, dolaylı

[https://en.wikipedia.org/wiki/List_\(abstract_data_type\)](https://en.wikipedia.org/wiki/List_(abstract_data_type))

Yazarlar, Katkı Sağlayanlar, ve Editörler (Grafikler Dahil)

Wilfried Baumann, author, baumann@ocg.at, Austria

Zoran Milevski, editor, milevskiz@gmail.com, North Macedonia

Madhavan Mukund, editor, madhavan@cmi.ac.in, India

Darija Dasović, graphics, darija.dasovic@ucitelji.hr, Croatia



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Dinle ve Yürü

Tolga, konuşan gözlüklerini kullanarak şehrin yollarında yürüyor. Konuşan gözlüklerde kameralar ve akıllı bir nesne tanıma sistemi vardır. Aşağıdaki haritanın dört tür karesini tanıyabilirler: bir ev, bir ağaç, bir yol ve çimen. Tolga yeni bir yol meydanına girdiğinde, konuşan gözlükler ona – bu sırayla – solunda ne olduğunu, önünde ne olduğunu ve sağında ne olduğunu söyler; örneğin: "ağaç yol evi".

Tolga üçgenden başlar (haritanın sağ tarafına bakar) ve konuşan gözlüklerini dinler. Ona söyledikleri bu (başlangıç karesinden başlayarak):

ağaç yol ev,

yol yol çim,

ağaç yol ağaç,

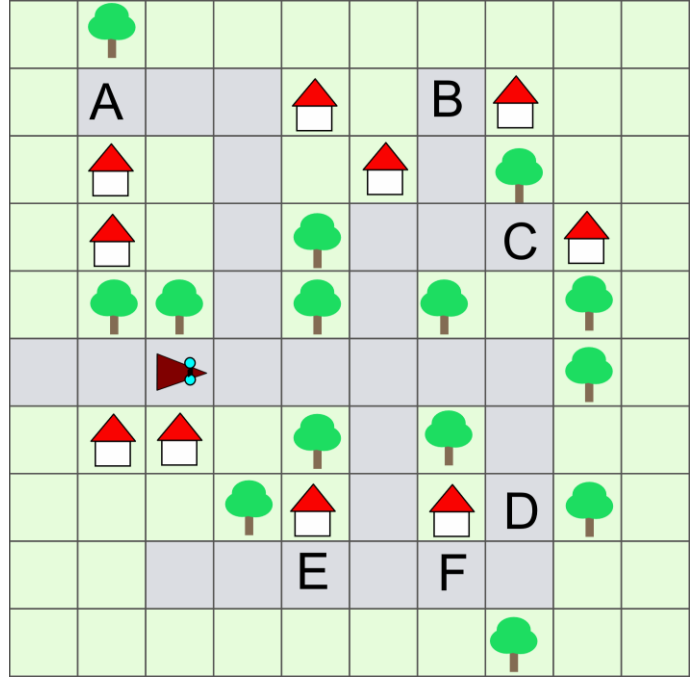
yol yol yol,

ağaç yol ağaç,

ağaç ev yolu,

yol yol ağacı,

ev yol ağacı



Sonunda Tolga bir harfle etiketlenmiş karelerden birine geldi.

Soru

Tolga, hangi kareye gelmiştir?

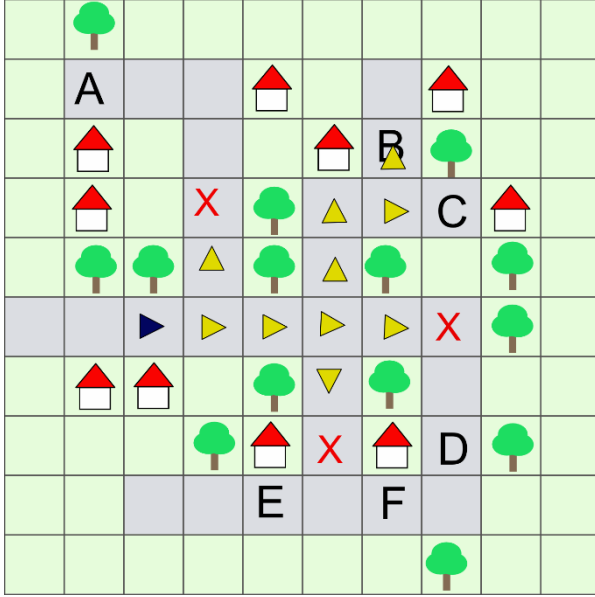
- A) F
- B) D
- C) C
- D) B



Cevap Açıklaması

YANIT D

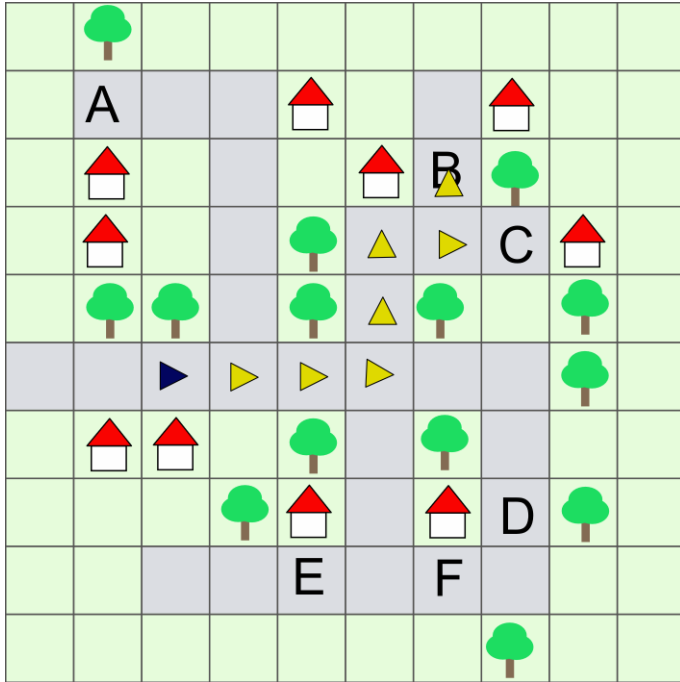
Çözümü bulmak için farklı stratejiler var. Baştan itibaren yolu takip edebilir ve konuşan gözlüklerin verdiği bilgilerle eşleşen patika boyunca yürüebilirsiniz.



Tina yolda bir dönüş yaptığında nesnelere bize söylemez. Bu nedenle, herhangi bir konumdaki bir sonraki nesne grubu, birden fazla yoldaki nesnelere eşleşirse, her bir yolu aynı anda deneriz. Örneğin, yol boyunca devam etmek için A hedefine giden yola gidersek, "ağaç yol ağacı"ndan sonraki (4.) nesne kümesinin "çim yol ağacı" olması gerektiğini fark ederiz. Ancak nesnelere, A varış noktasına giden yolda nesnelere eşleşmeyen "yol yolu yolu"dur. Böylece, A varış noktasına giden yolu seçtiğimiz kavşağa geri döneriz ve başka bir yol alırız. Bunu, olası tüm yolları deneyene veya hedeflerden birine ulaşana kadar yapıyoruz. Aşağıdaki

şekil Tina'nın alabileceği olası yolları göstermektedir. X, yola devam edemeyeceğini gösterir.

Alternatif olarak, altı hedef kareyi kontrol edebilirsiniz. Yalnızca B ve D, "ev yol ağacı" tanımıyla eşleşir. Ancak D'ye bitişik her iki kare de ikinci son tanım olan "yol yol ağacı" ile eşleşmiyor. Bu nedenle, B, Tina'nın hedef karesi olmalıdır.



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Enformatik Kavramı

Görev, konuşan gözlüklerin Tolga'ya söylediği mesajların bir günlüğünü bilmenin, birkaç olasılık arasından son karesini kurtarmak için ne kadar yeterli olduğunu gösteriyor. Genel olarak, bu günlüğün Tolga'nın şu anda nerede olduğunu söylemek için yeterince bilgilendirici olmadığı düşünülebilir, ancak hareket ettiği ortam hakkında sahip olduğumuz ek bilgi zengindir ve hedefini takip edebiliriz. Kişisel bilgilerin ne kadar hassas olabileceğini değerlendirmek her zaman zordur ve bu özellikle uzaysal veya coğrafi veriler için geçerlidir.

Nesne tanıma, bilgisayar bilimlerinde önemli bir konudur. Evler, ağaçlar ve yol yüzeyleri gibi nesnelere tanıma için bir yapay zeka sistemi eğitilebilir. Muhtemelen akıllı telefonunuzun kamerası yüzleri tanıyabilir ve bir çerçeve ile gösterir. Otonom bir araba, çevresindeki kişileri, diğer arabaları, trafik işaretlerini ve nereye gideceğine karar vermesinde önemli olan diğer nesnelere tanıması gerekir. Görme engelli kişilere yardımcı olmak için modern web medyası Otomatik Alternatif Metin (AAT) içerir. Bu teknoloji, görüntülerin sözlü açıklamalarını (alternatif metinleri) otomatik olarak oluşturmak için nesne tanımayı kullanır. "Konuşan gözlükler" hala bilim kurgu ama bu fikri hayata geçirmeye çalışan gerçek projeler bulunmaktadır.

Anahtar Kelimeler ve Web Sayfaları

Bilgisayarla görüş, nesne tanıma, nesne algılama, otomatik alternatif metin

https://en.wikipedia.org/wiki/Computer_vision

https://en.wikipedia.org/wiki/Object_detection

<https://tech.fb.com/artificial-intelligence/2021/01/how-facebook-is-using-ai-to-improve-photo-descriptions-for-people-who-are-blind-or-visually-impaired/>

Yazarlar, Katkı Sağlayanlar, ve Editörler (Grafikler Dahil)

Michael Weigend (author), mw@creative-informatics.de

Mattia MONGA - Vipul SHAH (2022-05-17)



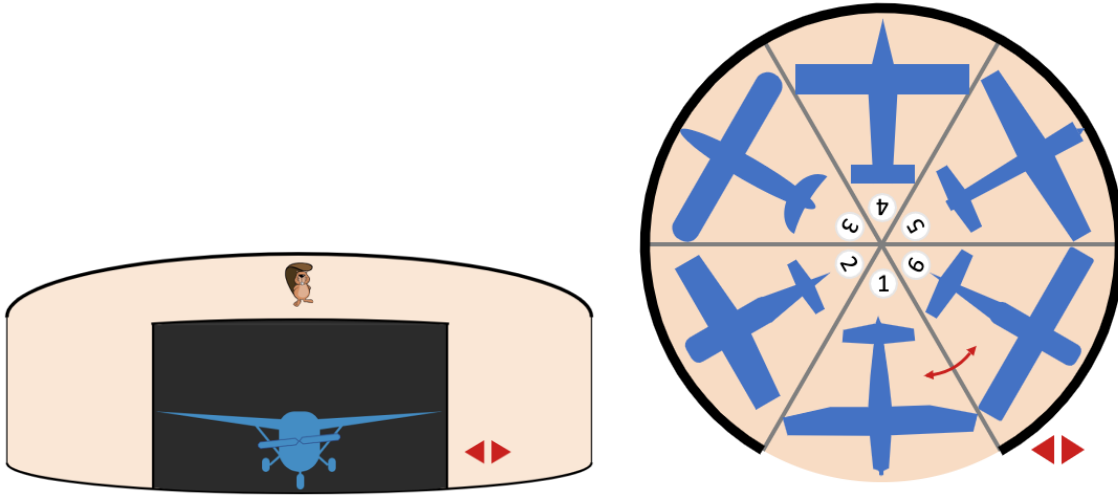
Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Döner Hangar

Atatürk havaalanında, yuvarlak hangarda dönen bir döner platform üzerine altı uçak park edilebilir. Döner tabla, ◀▶ iki oklu bir kontrol paneli kullanılarak sola veya sağa döndürülebilir. Bir düğmeye basılması, döner tablayı tam olarak sola veya sağa bir park pozisyonu döndürür. Hangarın kapısı, bir uçağın yuvarlanabileceği kadar geniştir. Döner tablanın dönmesi çok yavaştır, bu nedenle daha az düğmeye basmak gecikmeleri önleyecektir.



Sabahları, pilotlar uçaklarını almaya geldiklerinde, park yeri 1 her zaman kapıdadır. En iyi durumda, tüm uçakların fırlatılması için ok tuşlarına beş kez basılması gerekir. Bu durumlarda pilotlar park konumlarına sırayla: 1, 2, 3, 4, 5, 6 ▶ düğmesine beş kez basarak veya sırayla: 1, 6, 5, 4, 3, 2 ◀ düğmesine beş kez basarak erişmek isterler.

Soru

Park konumlarına erişim için aşağıdaki sıralardan hangisi tüm uçakların açılması için maksimum sayıda düğmeye basılmasını gerektirecektir?

- A) 4 1 3 6 2 5 ve 4 1 5 2 6 3
- B) 1 4 6 3 2 5 ve 4 1 5 2 6 3
- C) 4 1 3 6 2 5 ve 2 1 5 4 6 3
- D) 1 3 6 4 2 5 ve 2 4 1 5 6 3



Cevap Açıklaması

YANIT A

Maksimum sayıda düğmeye basılması için iki park pozisyonu sıralaması vardır: 4 1 3 6 2 5 ve 4 1 5 2 6 3. Dolayısıyla doğru cevap, kapıdan her zaman en uzak olan bir sonraki park yeri seçilerek bulunabilir. Buradaki zorluk, dönüşün sonucunu hayal etmek ve alan için alanın nasıl serbest bırakıldığını görselleştirmek. Döner tablayı her iki yönde de hareket ettirebildiğiniz için birden fazla doğru çözüm vardır; altı düzlemde iki doğru çözüm vardır.

4 1 3 6 2 5:

İlk olarak 4. konuma erişmek için üç kez basmanız gerekir (sol veya sağ)

Bir sonraki erişim konumu 1, üç kez basmayı gerektirir (sol veya sağ)

Sonraki erişim konumu 3, sağa iki kez basmayı gerektirir

Bir sonraki erişim konumu 6, üç kez basmayı gerektirir (sol veya sağ)

Sonraki erişim konumu 2, sağa iki kez basmayı gerektirir

Son olarak pozisyon 5'e erişmek için üç basma gerekir (sol veya sağ)

4 1 5 2 6 3:

İlk olarak 4. konuma erişmek için üç kez basmanız gerekir (sol veya sağ)

Bir sonraki erişim konumu 1, üç kez basmayı gerektirir (sol veya sağ)

Bir sonraki erişim konumu 5, sola iki basış gerektirir

Sonraki erişim konumu 2, üç kez basmayı gerektirir (sol veya sağ)

Bir sonraki erişim konumu 6, sola iki basma gerektirir

Son olarak pozisyon 3'e erişmek için üç basma gerekir (sol veya sağ)

Enformatik Kavramı

Yuvarlak hangar, uçakların çok fazla yerden tasarruf sağlayacak şekilde içine park edilebilmesi avantajına sahiptir. Peki ya zaman? Uçaklar park pozisyonlarını pilotların geliş sırasına göre kullanıyorsa, pilotlar zaman kaybetmeden hızlı bir şekilde birbiri ardına park edip kalkabiliyor. Uçakları alma işlemi için bu en iyi durumdur. Ancak bu Bebras görevinin gösterdiği gibi, çoğu süreç için en kötü durumlar da vardır. En kötü durumda, zaman kaybedilir, böylece uzay ve zaman arasında bir değiş tokuş olur.

Bilgisayar biliminde, bir sürecin verimliliğini değerlendirmek çok önemli bir rol oynar. Bir problem için en iyi durumları bilmek iyi olsa da, en kötü durum verimliliği karakterize etmek için daha önemlidir. Verimlilik, zaman, enerji ve depolama alanı gibi kaynakların ekonomik olarak kullanılması anlamına gelir. Günümüzde postayla sipariş işinde verimli depolamanın bir örneği uygulanmaktadır: Ürünlere verimli bir şekilde erişmek için, artık genellikle alfabeyle veya makale numarasına göre düzenlenmezler. Bunun yerine, örneğin malların genellikle ne sıklıkla satın alındığını hesaba katan karmaşık algoritmalarla göre ürünler istiflenirler.

Anahtar Kelimeler ve Web Sayfaları

Arama ve sıralama; en iyi durum, en kötü durum, ortalama durum; algoritmik verimlilik
<https://aircraft-carousel.com/gallery-/>

Yazarlar, Katkı Sağlayanlar, ve Editörler (Grafikler Dahil)

Kirsten Schlüter, kirsten.schlueter@sonnwendjoch.de, Germany (task proposal)

Editor, ITW 2022 WG O4: Taina Lehtimäki, taina@cs.nuim.ie, Ireland



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.
This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Favori Film

Bir grup arkadaş, yedi filmden hangisini izleyeceklerini seçmek istemektedir. Her arkadaş her filmi aşağıda gösterildiği gibi değerlendirir.

	1	2	3	4	5	6	7
Ada	✓	✓	✓	✓	✓	✓	✓
Nalan	●	✓	✓	●	●	✓	✓
Timuçin	✗	✗	✗	●	✗	✗	✗
Giray	✗	●	●	●	✗	●	✗
Eda	✓	●	✗	✗	●	✓	✓
Gül	●	✗	●	✗	✓	●	●

En iyiden en kötüye derecelendirmeler:



Tüm arkadaşlar bir filme en iyi derecesini verirse, o filme "favori film" denir. Örneğin 1. film favori bir film değil çünkü Timuçin en iyi derecesini 4. filme vermiştir.

Soru

Bir filmin, favori film olması için değiştirilmesi gereken en küçük olası derecelendirme sayısı nedir?

- A) 1 B) 2 C) 3 D) 4



ÇÖZÜM

YANIT B

Favori bir film olacak şekilde sadece 1 girişi değiştirmenin bir yolu yoktur. Ankette, her film için başka bir film en az iki arkadaş tarafından daha iyi derecelendirildi:

Film	Bu filmin favori olması için tercihini değiştirmesi gereken kişiler
1	4: Nalan, Timuçin, Giray, and Gül
2	3: Timuçin, Eda, and Gül
3	3: Timuçin, Eda, and Gül
4	3: Nalan, Eda, and Gül
5	3: Nalan, Giray, and Eda
6	2: Timuçin and Gül
7	3: Timuçin, Giray, and Gül

Şimdi Ada'nın, Timuçin ve Gül'ü 6. film favori bir film olacak şekilde her birini bir derece değiştirmeye ikna etmesi gerekiyor. Timuçin ve Gül, 6. film için reytinglerini yükseltmeliler. Alternatif olarak, 6'dan daha iyi oy verdikleri filmlerin reytinglerini düşürebilirler: yani film 4 (Timuçin) ve 5 (Gül). Bununla birlikte, Ada'nın yalnızca benzer gelecekteki durumlarda derecelendirme iyileştirmelerini dikkate alması gerektiğini görmek kolaydır.

Sorudaki Enformatik Kavramı

Görevi çözmek için ne yaptın? Bunun bir yolu, her filmi tek tek ve diğer filmlerden herhangi biri için daha iyi bir derecelendirme olup olmadığını her kişi için tek tek kontrol etmektir. Sonunda, yukarıda verilen gibi bir tabloya ulaşılmaktadır. O zaman, en az sayıda değişiklikle en sevdiğiniz filme ulaşmak için hangi kişilerin derecelendirmelerini değiştirmelerini isteyeceğinizi bulabilirsiniz. Ada, problemini etkili bir şekilde çözmek için bu algoritmayı kullanabilir.

Ancak algoritma da verimli mi? Yani, Ada daha hızlı yapabilir mi? Genelde M film ve F arkadaş vardır. Ada'nın tüm M*F tablo girişlerine bakması gerekecek ve her biri için aynı kişinin M - 1 diğer derecelendirmesine bakması gerekecek. Genel olarak, $M * (M - 1) * F$ derecelere bakması gerekecek. Ama aslında, bir kişinin belirli bir filme verdiği puanın bir sorun teşkil edip etmediğini görmek için Ada'nın sadece bu kişinin en iyi genel puanını bilmesi gerekiyor. Spesifik derecelendirme genel olarak en iyi olandan daha düşükse, derecelendirilen film favori film olamaz. Yani, Ada önce her bir kişinin genel en iyi puanlarını öğrenirse (tüm M * F puanlarına bakarak), tüm M * F puanlarının ilgili kişinin en iyi puanından düşük olup olmadığını kontrol edebilir. Sonuç olarak, en iyi derecelendirmelerin ön hesaplamalı bu alternatif algoritması, Ada'nın $2 * M * F$ derecelendirmesine bakmasını sağlar. M = 7 ve F = 6 ile bunlar 84 aramadır, önceki algoritma ise 252 arama içerir. İkinci algoritma da Ada'nın problemini etkili bir şekilde çözer, ancak ilkinden daha verimlidir.



Bilgisayar bilimcileri için temel faaliyetlerinden biri, bir sorunu yalnızca etkili bir şekilde değil, aynı zamanda mümkün olduğunca verimli bir şekilde çözmektir. Daha hızlı donanım, bilgisayarların sorunları daha hızlı çözmesini sağlar, ancak bir sorun için verimli bir algoritma yoksa hızlı donanımın pek bir faydası olmaz.

Anahtar Kelimeler

verimlilik, verimli algoritma, ön hesaplama

Yazarlar ve Katkı Sağlayanlar

- Wolfgang Pohl, pohl@bwinf.de, Germany: initial task proposal (April 1, 2022)
- WS Local WG1: Goran Sukovic, Zsuzsa Pluhár, pluharzs@ik.elte.hu, (2022-05-17)



Copyright © 2022 Bebras – International Challenge on Informatics and Computational Thinking.

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

<https://creativecommons.org/licenses/by-sa/4.0/>

Tic-Tac-Toe

Tic-tac-toe, iki oyuncu için bir kağıt-kalem oyunudur.

Kurallar:

Bir oyuncu başlar, ardından her iki oyuncu sırayla boşlukları X veya O ile üçe üç ızgarada işaretler. İşaretlerinden üçünü yatay, dikey veya çapraz sıraya yerleştirmeyi başaran oyuncu kazanır. Hiç kimse başarılı olmazsa ve dokuz kutunun tamamı doldurulursa, oyun berabere biter.

Örnek:

Aşağıdaki resimler bir oyunun ilk hamlelerini ve son hamlesini göstermektedir (her pozisyonun son hamlesi vurgulanmıştır):

		X			O		X			O		X			O		X			O		X
																O					O	O
										X				X						X	X	X

En sağdaki görüntüye tamamlanmış bir oyunun sonuç sayfası diyoruz. Rastgele "X" veya "O" ile doldurulan tüm sayfalar, yukarıda belirtilen kurallara göre geçerli sonuç sayfaları değildir.

Soru

Yukarıdaki kurallara göre tamamlanmış bir oyunun tek geçerli sonuç sayfası aşağıdaki resimlerden hangisidir?

A	B	C	D
X	X	X	X
O	O	X	O
O	X	O	X



Cevap Açıklaması

Doğru cevap: C

Cevap C doğrudur, çünkü oyun O oyuncusu tarafından kazanıldı ve ardından durduruldu.

Cevap A doğru değil. Oyuncu X oyunu kazandı ama O oyuncusu X oyuncusundan daha fazla puan aldı, bu mümkün değil. Kazanan oyuncu her zaman son işareti koyduğundan, yalnızca daha fazla veya eşit sayıda işarete sahip olabilir, ancak daha az olamaz.

Cevap B doğru değil çünkü 5 X işareti var ama sadece 3 O işareti var. Bu imkansız. Rakamlar arasındaki fark sadece 0 veya 1 olabilir.

Cevap D doğru değil çünkü bir kazanan belirlenmemiş ve alanlar tam olarak doldurulmamıştır.

Enformatik Kavramı

Makul olup olmadıklarına bakmaksızın dört resmi kontrol ederek görevi çözdük.

Oyunun kurallarından, örneğin aşağıdaki gibi bir sonuç sayfasının yapısı hakkında kurallar türetebilirsiniz:

- 1) O-işaretleri ile X-işaretlerinin sayıları arasındaki fark 0, 1 veya -1 olmalıdır.
- 2) Hiçbir oyuncu kazanmadıysa, tüm alanlar doldurulmalıdır.
- 3) Kazanan oyuncu, kaybeden oyuncuya eşit veya bir fazla puana sahip olmalıdır.
- 4) Bir sonuç sayfasında yalnızca bir kazanan sıralama olabilir.

Bir resim bu kurallardan biriyle çelişiyorsa geçerli bir sonuç sayfası olamaz.

Veri işleyen bilgisayar sistemlerinde kurallar çok önemlidir. Örneğin, görüntü dosyalarının biçimini, kredi kartı numaralarını ve hatta telefon numaralarını tanımlayan kurallar vardır. Bir resim düzenleyici ile bir dosyayı açmaya çalıştığınızda, yazılım önce bu dosyanın içeriğinin bir resim formatının kurallarına göre geçerli veri olup olmadığını kontrol eder.

Anahtar Kelimeler ve Web Sayfaları

Kural, Tic-Tac-Toe, Geçerlilik Kontrolü

Yazarlar, Katkı Sağlayanlar, ve Editörler (Grafikler Dahil)

Mārtiņš Opmanis, martins.opmanis@lumii.lv, Latvia.

Ilze Nilandere, ilze.nilandere@visma.com, Latvia.

Wilfried Baumann, baumann@ocg.at, Austria.

Michael Weigend, mw@creative-informatics.de, Austria.

