

# ODTÜ BİLGİSAYAR TOPLULUĞU ÜNİVERSİTE ÖĞRENCİLERİ ARASI VIII. GELENEKSEL PROGRAMLAMA YARIŞMASI ÖN ELEME SORULARI

Topluluğumuz, 1997'den bu yana geleneksel olarak düzenlediği programlama yarışması serisine bu sene sekizincisini ekliyor. Programlama yarışması, Ulusal Bilim Olimpiyatları formatında, C ve C++ dilleri üzerinden yapılan ve soruları bilgisayar bilimleri alanının temel problemlerinden ilham alan bir yarışmadır. Yarışmamız, dünyadaki benzerleri arasında (ACM, Tübitak, IOI, vs...) Linux platformunda düzenlenmiş yarışmaların ilki olma ayrıcalığına sahiptir. Ön eleme sorularının son gönderim tarihi 15 Nisan 2005'tir. Ön katılımcılar arasından bu sorular yoluyla belirlenecek yaklaşık 20 finalist, 24 Nisan 2005 tarihinde ODTÜ Bilgisayar Mühendisliği Bölümü'nde düzenlenecek olan finale çağrılacaktır. Özel ödüllü soruyu en iyi çözen yarışmacı ve final sonucunda ilk üç dereceyi alan finalistler; ödülleri akşam ODTÜ Kültür ve Kongre Merkezi'nde düzenlenecek olan törende alacaklardır. Yarışma ile ilgili duyurularımızı web sitemizden takip edebilirsiniz. web sitesi: [yarisma.cclub.metu.edu.tr](http://yarisma.cclub.metu.edu.tr)  
e-mail: [yarisma@cclub.metu.edu.tr](mailto:yarisma@cclub.metu.edu.tr)

yarisma.cclub.metu.edu.tr is provided. The background is blue with white and yellow geometric shapes, including a large yellow arrow pointing right. There are binary code patterns and a small penguin mascot at the bottom left."/>

ODTÜ BİLGİSAYAR TOPLULUĞU

## VIII. ÜNİVERSİTE ÖĞRENCİLERİ ARASI GELENEKSEL PROGRAMLAMA YARIŞMASI

- C / C++ DİLLERİNDE
- LINUX ORTAMINDA
- FİNAL 24 NİSAN'DA
- ÖN ELEME SORULARI VE ÖDÜLLER İÇİN

[yarisma.cclub.metu.edu.tr](http://yarisma.cclub.metu.edu.tr)

0 1 0 1 1 1 0 1 1 1  
0 1 1 1 0 1 1 1 0 1  
1 1 1 1 1 1 1 1 0 1  
0 1 0 1 1 1 0 1 1  
1 0 1 1 0 1 1 0 1



## Kabile

Kahramanımız Atasay, bir sabah uyandı-ğında kendisini bir adada bulur. Buraya nasıl geldiğini hatırlayamayan Atasay'ı, adada yaşayan ilkel bir kabilenin muhafızları tutuklayarak kabile şefinin yanına götürürler. O sırada, kabilenin yaşadığı köyde, yaklaşan kutsal Burak Günü için hazırlıklar sürmektedir. Bu kutsal günde kabileden kurayla seçilen iki kişi arasında oynanan oyun sonunda kaybeden taraf tanrıya kurban edilir. Kabile şefi Ali Galip, Atasay'ın ancak oyuna girip kazandığı takdirde serbest bırakılacağını söyler. Ve Burak Günü gelir... Kabileden kurayla seçilen Yiğit, Atasay ile birlikte bir labirente getirilir ve oyun kuralları açıklanmaya başlanır.

Dikdörtgen şeklindeki labirent, kare odalardan oluşmaktadır. Boş odalar koridorları,

dolu odalar duvarları oluşturmaktadır. Boş odalardan bazılarında kutsal Bartan kuşunun tüyleri bulunmaktadır. Oyunculara, tüylerin yerini de gösteren labirent haritası ve oyuncu sırası verilir. Oyuna birinci oyuncu başlar ve sırası gelen oyuncu kuzey, güney, doğu ve batı yönlerindeki dört komşu odadan boş olan birisine geçer. Oyundaki en son tüyün alındığı hamlede, en fazla tüy toplayan oyuncu Bartan kuşu tarafından labirentten kurtarılır. Daha sonra labirent, bir ayınle, kalan oyuncu için deyen yakılır.

### Varsayımlar

Labirentin boyutları  $n \times m$  'dir.  $n$  satır sayısını,  $m$  sütun sayısını ifade etmektedir ( $2 \leq n, m \leq 30$ ).

İlk oyuncu oyuna labirentin kuzey batı köşesinden, ikinci oyuncu ise güney doğu

köşesinden başlayacaktır.

Bir oyuncunun diğerinin bulunduğu odaya gitmesinde (iki oyuncunun aynı anda aynı odada bulunmasında) bir sakınca yoktur.

Programınız ilk olarak "kabile.gir" isimli dosyadan labirent bilgisini okumalıdır. Daha sonra standart girdiden (stdin) oyuncu numarasını okuyarak oyuna başlamalıdır. Oyun esnasında, sıra kendisinde ise standart çıktıya (stdout) hamlesini basmalı, sıra rakipte ise standart girdiden rakibin hamlesini okumalıdır.

### Girdi-Çıktı

#### kabile.gir:

Girdi dosyası *kabile.gir* 'in ilk satırında labirentin boyutlarını gösteren  $n$  (satır sayısı) ve  $m$  (sütun sayısı) tamsayıları bulunacaktır. Takip eden  $n$  adet satırın her birinde aralarında

## Arthur

Kral Arthur, bir gün baş büyücüsü Umud'u yanına çağırarak ondan ölümsüzlük iksiri yapmasını ister ve yapamadığı takdirde kendi yaptığı zehirle öldürüleceğini söyler. Büyücü Umud çaresiz bir şekilde başlar çalışmaya... Günler süren yoğun çalışmadan sonra yorgunluğa dayanamaz ve uykuya dalar. Rüyasında kendisini bir anda ucu bucağı görünmeyen sonsuz ve karelere ayrılmış bir tarlada görür. Tarlanın karelere ayrılmış dikdörtgen şeklindeki bir bölümünde, her karede bir şişe olmak üzere iksir şişelerinin yerleştirilmiş olduğunu görür. Bir anda ak sakallı bir dede belirir ve Umud'a der ki: "Bir iksiri diğerinin üzerinden atlatırsan, atlattığın iksir diğerini yok eder ve tüm etkiyi kendisinde toplar. Sonuçta geriye en az sayıda iksir kalmasına gayret et ve kalan bu iksirleri Arthur'a götür. Hepsinden birer bardak içerse artık ölümsüzdür.". Bunları söyleyen ak sakallı dede bir anda kaybolur ve Umud uyanır. Hemen alır eline kağıt kalemi ve düşünmeye başlar. Fakat çok da vakti kalmamıştır. Sizden istenen Umud'a yardımcı olmanız ve rüyasında gördüğü problemi çözmeniz.

### Varsayımlar

- Dikdörtgen şeklindeki bölüm yatayda  $m$ , dikeyde  $n$  kareden oluşmaktadır ( $1 \leq n, m \leq 1000$ ).
- İksirler başka bir iksirin üzerinden atlama hareket ettirilemez. Sadece komşularının (kuzey, güney, doğu ve batısındaki dört kare) üzerinden atlatılabilir ve atladıktan sonra kolulacağı kare boş olmalıdır. Atlatılan iksir dik-

birer boşluk bulunan  $m$  adet tamsayı (0, 1 ya da 2, sırayla bütün odalar için, boş odalar için 0, dolu odalar için 1, tüy bulunan odalar için 2) bulunacaktır.

standart girdi - standart çıktı:

Programınız ilk olarak standart girdiden sizin programınızın oyuncu numarasını belirten bir adet tamsayı okuyacaktır. Bu sayı birinci oyuncu için 1, ikinci oyuncu için 2 olacaktır.

İlerleyen aşamalarda, hamle sırası karşıdaki oyuncuda ise standart girdiden onun yaptığı hamleyi okuyacak, sıra kendisinde ise standart çıktıya kendi hamlesini yazacaktır. Hamle, 'K' (kuzey), 'G' (güney), 'D' (doğu), 'B' (batı) karakterlerinden birisi (1 adet karakter) ve takip eden satır sonu karakterinden ('\n') oluşmalıdır.

### Değerlendirme

Yazdığımız kod, verilen her labirent için gönderilen bütün diğer kodlar ile hem 1. hem de 2. oyuncu için oynayacaktır.

Herhangi bir anda yanlış bir hamle yapan oyuncu o oyunu kaybetmiş sayılacaktır. Yanlış hamle, kapalı bir odaya gitmeye çalışmak, la-

dörtgen şeklindeki bölümün dışına çıkabilir.

• Başlangıçta, dikdörtgen şeklindeki bölümün bütün karelerinde tam olarak birer iksir vardır.

• Sonuçta geriye en az sayıda iksir bırakmak gerekmektedir.

• Dikdörtgen şeklindeki bölümün kuzey-batı köşesinin koordinatları (0,0) kabul edilmelidir. Koordinatlar (satır, sütun) şeklinde gösterilmektedir.

• Programınız girdileri "arthur.gir" isimli dosyadan okuyup, çıktılarını "arthur.cik" isimli dosyaya yazmalıdır.

### Girdi (arthur.gir)

Girdi dosyası arthur.gir'in ilk satırında iki adet tamsayı, sırayla  $n$  ve  $m$  aralarında bir adet boşluk bulunacak şekilde verilecektir.

İlk durum:	İlk hamle sonunda:	İkinci hamle sonunda:																																																																				
<table border="1"> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> <tr><td>x</td><td>x</td><td>x</td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> <tr><td>x</td><td>x</td><td>x</td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> </table>													x	x	x						x	x	x						<table border="1"> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td>x</td><td></td><td></td></tr> <tr><td></td><td></td><td>x</td><td>x</td></tr> <tr><td></td><td></td><td>x</td><td>x</td></tr> <tr><td></td><td></td><td></td><td></td></tr> </table>						x					x	x			x	x					<table border="1"> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td>x</td><td>x</td><td></td></tr> <tr><td></td><td></td><td></td><td>x</td></tr> <tr><td></td><td></td><td></td><td>x</td></tr> <tr><td></td><td></td><td></td><td></td></tr> </table>						x	x					x				x				
x	x	x																																																																				
x	x	x																																																																				
	x																																																																					
		x	x																																																																			
		x	x																																																																			
	x	x																																																																				
			x																																																																			
			x																																																																			
Üçüncü hamle sonunda:	Dördüncü hamle sonunda:																																																																					
<table border="1"> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td>x</td><td>x</td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> </table>						x	x										<table border="1"> <tr><td></td><td></td><td></td><td></td></tr> <tr><td>x</td><td></td><td></td><td>x</td></tr> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> </table>					x			x									<table border="1"> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> <tr><td></td><td></td><td></td><td></td></tr> </table>																																				
	x	x																																																																				
x			x																																																																			

K  
B ← ↑ → D  
G

birentin dışına çıkmaya çalışmak, yanlış bir karakter basmak vb. olabilir.

Oyuncuların kodları, bizim yazacağımız bir hakem kodu aracılığıyla oynatılacaktır.

### Örnek:

kabile.gir:  
5 6  
0 0 1 0 2 1  
0 1 2 0 0 0  
0 2 1 0 1 0  
0 0 1 0 0 0  
1 0 2 0 0 0

P1				X	
		X			
	X				
		X			P2

Labirentin ilk durumu:  
P1: Birinci oyuncu  
P2: İkinci oyuncu  
X: Tüy  
gri kareler: Dolu odalar

K  
B ← ↑ → D  
G

## Çıktı (arthur.cik)

Programınız arthur.cik dosyasında sırasıyla yaptığımız hamleler bulunmalıdır. Her satırda bir hamleniz yer almalıdır. Bir hamle aralarında birer boşluk bulunan 4 adet tamsayı içermelidir. İlk iki sayı atlatılan iksirin koordinatını, üçüncü ve dördüncü sayılar ise üzerinden atlanan iksirin koordinatını göstermelidir.

### Örnek

arthur.gir:  
2 3  
arthur.cik  
1 0 0 0  
1 1 0 1  
1 2 0 2  
-1 1 -1 0

Daha sonra birinci oyuncunun standart inputuna '1', ikinci oyuncunun standart inputuna '2' yazılır. Oyuncular bu sayıyı okuduktan sonra birinci oyuncu:

G

gibi bir hamle yapar (standart çıktıya). İkinci oyuncu birincinin hamlesini okuyarak kendi hamlesini basar (standart çıktıya) ve oyun bütün tüyler toplanana kadar devam eder. Oyunun sonlandırılması, birinci oyuncunun standart çıktısının ikinci oyuncunun standart girdisini beslemesi veya tam tersi hakemin görevidir. Oyuncunun bunun için bir şey yapmasına gerek yoktur.

## Kabile -Yıllar Sonra- (Özel Soru)

NOT: Bu sorunun hikayesi, *Kabile* sorusunun hikayesinin devamı niteliğindedir. Öncelikle o soruyu okumanızı tavsiye ederiz. (Bu soruyu çözmek için *Kabile* sorusunu çözmüş olmanız gerekmemektedir.)

Kahramanımız Atasay, hafızasını kaybetti-

ği için bu adaya nereden ve nasıl geldiğini hatırlamamaktadır. Burak Günü'nde de hayatı kurtulduğu için köye yerleşmeye karar verir ve burada yaşamaya başlar. Yıllar sonra yine bir Burak Günü'nde, kura Atasay'a çıkar ve kura-da seçilen diğer köylü Oğuz'la birlikte labirente götürülür. Kabile şefi Ali Galip öldükten sonra yerine geçen oğlu Gökdeniz, oyuna yeni bir kural eklemiştir: Bir oyuncu, sıra kendisinde iken komşu odalardan boş olan birisini bir daha açılmamak üzere kapatabilir (içinde diğer oyuncu bulunan kare kapatılamaz, tüy bulunan kare kapatılabilir). Yani bir oyuncu hamle hakkını, ya boş olan komşu bir odaya geçerek, ya da yerinde kalıp boş olan komşu bir odayı kapatarak kullanır.

### Varsayımlar

Labirentin boyutları  $n \times m$ 'dir.  $n$  satır sayısını,  $m$  sütun sayısını ifade etmektedir ( $2 \leq n, m \leq 30$ ).

İlk oyuncu oyuna labirentin kuzey batı köşesinden, ikinci oyuncu ise güney doğu köşesinden başlayacaktır.

Bir oyuncunun diğerinin bulunduğu odaya gitmesinde (iki oyuncunun aynı anda aynı odada bulunmasında) bir sakınca yoktur.

Programınız ilk olarak "kabile2.gir" isimli dosyadan labirent bilgisini okumalıdır. Daha sonra standart girdiden (stdin) oyuncu numarasını okuyarak oyuna başlamalıdır. Oyun esnasında, sıra kendisinde ise standart çıktıya (stdout) hamlesini basmalı, sıra rakipte ise standart girdiden rakibin hamlesini okumalıdır.

### Girdi-Çıktı

#### kabile2.gir:

Girdi dosyası *kabile2.gir*'in ilk satırında labirentin boyutlarını gösteren  $n$  (satır sayısı) ve  $m$  (sütun sayısı) tamsayıları bulunacaktır. Takip eden  $n$  adet satırın her birinde aralarında birer boşluk bulunan  $m$  adet tamsayı (0, 1 ya da 2, sırayla bütün odalar için, boş odalar için 0, dolu odalar için 1, tüy bulunan odalar için 2) bulunacaktır.

#### standart girdi - standart çıktı:

Programınız ilk olarak, standart girdiden sizin programınızın oyuncu numarasını belirten bir adet tamsayı okuyacaktır. Bu sayı birinci oyuncu için 1, ikinci oyuncu için 2 olacaktır.

İlerleyen aşamalarda, hamle sırası karşındaki oyuncuda ise standart girdiden onun yaptığı hamleyi okuyacak, sıra kendisinde ise standart çıktıya kendi hamlesini yazacaktır. Hamle, hareketler için 'HK' (kuzey), 'HG' (güney), 'HD' (doğu), 'HB' (batı) veya oda kapatmak için 'OK' (kuzey), 'OG' (güney), 'OD' (doğu), 'OB' (batı) karakter çiftlerinden birisinden (2 adet

karakter) ve takip eden satır sonu karakterinden ('\n') oluşmalıdır.

### Değerlendirme

• Yazdığınız kod, verilen her labirent için gönderilen bütün diğer kodlar ile hem 1. hem de 2. oyuncu için oynayacaktır.

• Herhangi bir anda yanlış bir hamle yapan oyuncu o oyunu kaybetmiş sayılacaktır. Yanlış hamle, kapalı bir odaya (sonradan da kapatılmış olabilir) gitmeye çalışmak, kapalı bir odayı tekrar kapatmaya çalışmak, rakip oyuncunun olduğu odayı kapatmaya çalışmak, labirentin dışına çıkmaya çalışmak, yanlış bir karakter basmak vb. olabilir.

• Oyuncuların kodları, bizim yazacağımız bir hakem kodu aracılığıyla oynatılacaktır.

## Midas'ın Kareleri



Kral Midas, bir gün bahçede dolaşırken bir kağıt bulur. Kağıdın üzerinde bir harita ve açıklayıcı bir metin vardır, fakat kağıttaki tüm yazılar şifrelenmiştir. Şifreli yazıyı çözemeyen Midas, kağıdı ulu bilge Yaba'ya götürür. Şifreyi çözen Yaba, Midas'a haritada belirtilen adreste, zemini karelerle döşenmiş bir oda ve odadaki her karede de altınların olduğunu söyler. Odanın duvarları camdan olduğu için odadaki altın dağılımı odaya girmeden önce görülebilmektedir ve de istenilen kareden odaya girmek mümkündür. Odaya giriş şifrelidir ve şifre odadan toplanabilecek maksimum altın sayısına eşittir. Midas altınları riske etmemek için ülkede bir yarışma başlatır. Şehrin her yerine odadaki altın dağılımını gösteren haritalar asılır ve odadan çıkarılabilecek maksimum altın sayısını bulan kişi Midas'ın kızıyla evlenme hakkını kazanacaktır.

Ancak yarışmayı kazanan kişi, eğer odadan bulunduğu sayıdan daha az sayıda altınla çıkarsa çivili fiçiyi atılacaktır. Yıllardır Midas'ın kızını seven Başkumandan Loga da yarışmak zorunda olduğunu anlar ve yardımınıza başvurur. Bakalım yazacağınız program ona yardım edebilecek mi?

### Odadaki Hareket Kuralları:

- Oyuncu odaya istediği kareden girer (içteki kareler de dahil).

- Oyuncu bulunduğu kareden sadece kuzey, güney, doğu ve batı yönlerindeki 4 kareden birine gidebilir.

- Oyuncu her geçişte bulunduğu karedeki altınları alır. Oyuncu kareden ayrıldıktan sonra altınlar yeniden doğmaktadır (Diğer şartlar sağlandığı sürece tekrar tekrar aynı kareye geline-

rek altın alınabilir).

- Oyuncunun bir kareden başka bir kareye gidebilmesi için, gideceği karedeki altın sayısı bulunduğu kareden az olmalıdır. Oyuncu ancak sınırlı sayıda joker haklarından her seferinde birisini kullanarak bu kuralı jokerleri tükeninceye kadar çiğneyebilir.

-Oyuncunun kullanabileceği joker sayısı odanın duvarında yazmaktadır.

### Varsayımlar

Odanın kare cinsinden boyutları  $n \times m$ 'dir.  $n$  satır sayısını (boyunu),  $m$  sütun sayısını (eni) ifade etmektedir ( $2 \leq n, m \leq 1000$ ).

Kullanılabilecek maksimum joker sayısı  $k$ 'dir ( $k \leq 30$ ).

Programınız girdileri "midas.gir" isimli dosyadan okuyup, çıktılarını "midas.cik" isimli dosyaya yazmalıdır.

### Girdi (midas.gir)

Girdi dosyası *midas.gir*'in ilk satırında bir adet tamsayı  $k$ , ikinci satırında ise iki adet tamsayı  $n$  ve  $m$  aralarında bir adet boşluk bulunacak şekilde verilecektir. Üçüncü satırından itibaren takip eden  $n$  adet satırda ise kuzey-batı köşesinde bulunan kareden başlayarak odada bulunan her karedeki altın sayısı (her satırda  $m$  adet pozitif tamsayı aralarında birer boşluk olacak şekilde) verilecektir.

### Çıktı (midas.cik)

Programınız *midas.cik* dosyasına odadan toplanabilecek maksimum altın sayısını yazacaktır.

### Örnek :

```
midas.gir:
1
3 4
2 5 8 9
7 6 9 10
1 1 8 19
```

```
midas.cik:
99
```

2	5	8	9
7	6	9	10
1	1	8	19

En fazla altının toplandığı yol:  $B \leftarrow \begin{matrix} K \\ \updownarrow \\ G \end{matrix} \rightarrow D$   
 $19-10-9(2.\text{satır}, 3.\text{sütun})-19-10-9(1.\text{satır}, 4.\text{sütun})-8(1.\text{satır}, 3.\text{sütun})-5-2$   
 2'nin bulunduğu karedeyken yapılabilecek hamle kalmadığı için program sonlandırılır.

Joker hakkı 8'den 19'a geçerken kullanılmıştır.