

[Dashboard](#) / [My courses](#) / [ITB IF2110 1 2324](#) / [Praktikum 4 \(18 September - 22 September\)](#) / [ADT Matrix - Praktikum \(extended\)](#)

Started on	Thursday, 30 November 2023, 3:45 PM
State	Finished
Completed on	Friday, 1 December 2023, 11:16 PM
Time taken	1 day 7 hours
Grade	55.00 out of 120.00 (46%)

Question **1**

Correct

Mark 40.00 out of 40.00

Time limit	1 s
Memory limit	64 MB

Diberikan matriks 8x8 yang menyatakan kondisi sebuah papan catur. Bila suatu elemen pada matriks bernilai 1 maka terdapat bidak catur pada sel tersebut, sedangkan bila bernilai 0 maka tidak terdapat bidak catur pada sel tersebut. Sebuah sel pada papan catur dapat berwarna putih atau hitam, sehingga setiap sel yang saling bersisian pasti memiliki warna yang berbeda. Diketahui bahwa sel paling atas kiri (koordinat = (row 1, col 1)) berwarna hitam. Tentukan jumlah bidak yang berada pada sel hitam dan jumlah bidak yang terdapat pada sel putih. Kumpulkan file **catur.c**

Format masukan

Matriks 8x8 yang dijamin hanya memiliki elemen bernilai 0 atau 1

Format keluaran

Jumlah bidak pada sel yang berwarna hitam dan jumlah bidak pada sel yang berwarna putih dengan dipisahkan spasi

Contoh masukan 1

```
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 0
0 1 0 0 0 0 1 0
1 0 0 0 0 0 0 0
```

Contoh keluaran 1

1 3

Penjelasan 1

Lokasi bidak pada sel hitam = ({row 7, col 7})
Lokasi bidak pada sel putih = ({row 8,col 1},{row 7, col 2},{row 6, col 7})

Note : Jangan lupa untuk memberikan endline setelah melakukan output

C

 [catur.c](#)

Score: 25

Blackbox

Score: 25

Verdict: Accepted

Evaluator: Exact

No	Score	Verdict	Description
1	2.5	Accepted	0.00 sec, 1.62 MB

No	Score	Verdict	Description
2	2.5	Accepted	0.00 sec, 1.66 MB
3	2.5	Accepted	0.00 sec, 1.65 MB
4	2.5	Accepted	0.00 sec, 1.66 MB
5	2.5	Accepted	0.00 sec, 1.73 MB
6	2.5	Accepted	0.00 sec, 1.68 MB
7	2.5	Accepted	0.00 sec, 1.64 MB
8	2.5	Accepted	0.00 sec, 1.63 MB
9	2.5	Accepted	0.00 sec, 1.66 MB
10	2.5	Accepted	0.00 sec, 1.75 MB

Question **2**

Partially correct

Mark 15.00 out of 30.00

Time limit	1 s
Memory limit	64 MB

Terdapat papan puzzle berukuran NxM yang sudah hampir selesai. Tuan Haka ingin menyelesaikan puzzle tersebut, namun sisa potongan-potongan puzzle-nya saling menempel karena adik Tuan Haka iseng mengelemnya.

Karena itu, Tuan Haka harus meletakkan K gabungan potongan puzzle berukuran AxB. Gabungan potongan tersebut bisa diletakkan dengan mencocokkan isi masing-masing potongannya dengan slot pada puzzle.

Puzzle dikatakan bisa diselesaikan bila setiap potongan puzzle merupakan submatriks dari puzzle. Potongan puzzle juga **tidak bisa dirotasi**. Tentukanlah apakah Tuan Haka bisa menyelesaikan puzzle tersebut! Kumpulkan dalam file **puzzle.c**.

Catatan : Antar potongan puzzle dijamin tidak beririsan

Format

Format Masukan	Format Keluaran
<u>N</u> <u>M</u> <i>Matriks NxM</i> <u>K</u> { <u>A</u> <u>B</u> <i>Matriks AxB</i> } (sejumlah K dengan A < N, B < M)	<ul style="list-style-type: none">"Puzzle dapat diselesaikan.""Puzzle tidak dapat diselesaikan."

Contoh

Masukan	Keluaran	Keterangan
<div><div>4 5</div><div>1 2 3 4 5</div><div>6 7 8 9 0</div><div>0 9 8 7 6</div><div>5 4 3 2 1</div><div>3</div><div>1 3</div><div>0 9 8</div><div>2 2</div><div>2 3</div><div>7 8</div><div>3 2</div><div>4 5</div><div>9 0</div><div>7 6</div></div>	<div>Puzzle dapat diselesaikan.</div>	<div>Terdapat 3 gabungan potongan yang ketiganya terdapat pada papan puzzle</div> <ul style="list-style-type: none">Gabungan potongan pertama: 1x3 0 9 8Gabungan potongan kedua: 2x2 2 3 7 8Gabungan potongan ketiga: 3x2 4 5 9 0 7 6 <div><div>1</div><div>2 3</div><div>4 5</div><div>6</div><div>7 8</div><div>9 0</div><div>0 9 8</div><div>7 6</div><div>5 4 3</div><div>2 1</div></div>

4	5			
1	2	3	4	5
6	7	8	9	0
0	9	8	7	6
5	4	3	2	1
2				
1	5			
5	4	3	2	1
2	3			
1	2	3		
4	5	6		

Puzzle tidak dapat diselesaikan.

Tidak terdapat pada papan puzzle

Dengan demikian puzzle tidak dapat diselesaikan

C

5/9

Question **3**

Not answered

Marked out of 30.00

Time limit	1 s
Memory limit	64 MB

Diberikan dua buah bilangan N dan M serta matriks dengan ukuran N x M (N baris dan M kolom). Diberikan pula sebuah bilangan K, Tentukan nilai maksimum dari total elemen pada segi empat (tidak harus persegi, bisa persegi panjang) yang memiliki luas sebesar K. Bila tidak terdapat segi empat yang memiliki luas sebesar K maka keluarkan nilai 0. Kumpulkan file **jumlahmaks.c**

Format masukan

N M K (N,M<= 10 dan K <= 100)

Matriks N x M

Format keluaran

Sebuah bilangan yang menyatakan nilai maksimum dari total elemen pada segi empat yang memiliki luas sebesar K

Contoh masukan 1

5 5 8

1 1 5 0 0

1 1 4 1 0

0 0 0 0 0

0 0 0 3 0

0 0 0 1 0

Contoh keluaran 1

14

Penjelasan 1

Segi empat dengan luas 8 yang memiliki total elemen terbesar yaitu

1 1 5 0

1 1 4 1

Total = 1+1+5+0+1+1+4+1 = 14

Dapat dibuktikan 14 merupakan nilai maksimum yang dapat diperoleh, adapun segi empat lainnya yang memiliki luas 8 yaitu

1 0

0 0

3 0

1 0

Total = 1+0+0+0+3+0+1+0 = 5 (tidak maksimum)

Contoh masukan 2

5 6 11

1 1 5 0 0 1

1 1 4 1 0 2

0 0 0 0 0 0

0 0 0 3 0 3

0 0 0 1 0 0

Contoh keluaran 2

0

Penjelasan 2

Tidak dapat dibentuk segi empat dengan luas 11 pada matriks 5 x 6

Note : Jangan lupa untuk memberikan endlime setelah melakukan output

C

Question **4**

Not answered

Marked out of 20.00

Time limit	1 s
Memory limit	64 MB

Soal ini bersifat Bonus

Fibonacci merupakan sebuah deret bilangan yang memenuhi $f(n) = f(n-1)+f(n-2)$, dengan **f(1) bernilai 1 dan f(2) bernilai 1**. Salah satu perhitungan fibonacci yang paling efisien yaitu dengan menggunakan matriks, berikut contohnya:

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \times \begin{bmatrix} f(2) \\ f(1) \end{bmatrix} = \begin{bmatrix} f(2) + f(1) \\ f(2) \end{bmatrix} = \begin{bmatrix} f(3) \\ f(2) \end{bmatrix}$$

Dengan mengalikan matriks $\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$ dengan $\begin{bmatrix} f(2) \\ f(1) \end{bmatrix}$ maka dapat dihasilkan $\begin{bmatrix} f(3) \\ f(2) \end{bmatrix}$. Perhatikan pula misalkan dikalikan matriks $\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$ lagi maka dapat dihasilkan $\begin{bmatrix} f(4) \\ f(3) \end{bmatrix}$

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \times \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \times \begin{bmatrix} f(2) \\ f(1) \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \times \begin{bmatrix} f(3) \\ f(2) \end{bmatrix} = \begin{bmatrix} f(4) \\ f(3) \end{bmatrix}$$

Sehingga dapat disimpulkan dengan mengalikan matriks $\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}$ secara terus menerus (biasa disebut sebagai perpangkatan) dapat dihasilkan bilangan fibonacci selanjutnya

$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{k-2} \times \begin{bmatrix} f(2) \\ f(1) \end{bmatrix} = \begin{bmatrix} f(k) \\ f(k-1) \end{bmatrix}$$

Oleh karena itu untuk menghasilkan bilangan fibonacci ke-k kita hanya perlu mendapatkan nilai dari matriks yang dipangkatkan k-2. Perpangkatan matriks yang efisien dapat dilakukan secara rekursif dengan cara berikut dalam bentuk pseudocode:

```
Matrix PangkatMatrix (Matrix m, int pangkat)
{
    if (pangkat == 0) return identityMatrix2x2
    else if (pangkat == 1) return m
    else
    {
        Matrix m2 = PangkatMatrix(m,pangkat/2)
        Matrix mulMatrix = multiplyMatrix(m2,m2)
        if (pangkat % 2 == 1)
        {
            mulMatrix = multiplyMatrix(mulMatrix,m)
        }
        return mulMatrix
    }
}
```

Tentukan bilangan fibonacci ke-k dengan **dimodulo 2003**. Gunakan fungsi **multiplyMatrixWithMod** saat mengalikan matriks. Bila kalian mendapat verdict Time Limit Exceeded pada olympia berarti implementasi perpangkatan matriks yang kalian buat masih belum efektif. Kumpulkan file **fibonacci.c**

Format masukan

K (K <= 10^9)

Format keluaran

Bilangan Fibonacci ke-K yang dimodulo 2003

Contoh masukan 1

3

Contoh keluaran 1

2

Contoh masukan 2

1000000000

Contoh keluaran 2

1982

Note : Jangan lupa untuk memberikan endline setelah melakukan output

C

◀ ADT Matrix - Praktikum

Jump to...

[boolean.h](#) ▶