

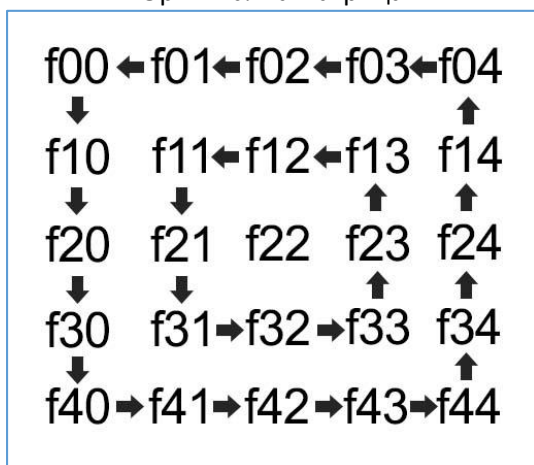
- Сваки задатак носи максимално 50 бодова.
- На почетку сваког задатка уносе се потребне улазне вредности.
- Сваки задатак има 5 излаза који се рачунају на основу улазних вредности унетих на почетку.
- Сваки излаз у задатку носи 10 бодова ако је тачан, 0 ако је нетачан.
- Такмичари су обавезни да **ЈАСНО** назначе излазе (погледати слику на примерима). Ако излаз није јасно назначен такмичар добија 0 поена.
- Ако задатак престане да ради због грешке у било ком тренутку, такмичару се бодују излази који су до тог тренутка видљиви на екрану.
- Такмичење траје 180 минута.

Задатак 1

Унети 10-цифрени број a ($1000000000 < a < 2000000000$) и број ротирања z . Претпоставка је да је број исправно унет. Нема потребе проверавати да ли је унос заиста 10-цифрени број.

- 1.1 Прекопирати број a у број b . Израчунати и приказати средњу вредност цифара броја b .
- 1.2 Прекопирати број a у број c . Од броја c направити 5 двоцифрених бројева тако што ће се користити 1. и 10. цифра за 1. двоцифрени број, 2. и 9. цифра за 2. двоцифрени број, итд..... Од добијених бројева приказати само просте.
- 1.3 Од броја a направити низ d од 10 елемената (индексирање низа почиње од 0) на следећи начин: 1. елемент низа је број понављања цифре 0 у броју a , 2. елемент низа је број понављања цифре 1 у броју a , итд..... Уколико се цифра појављује 0 пута, елемент низа на тој позицији је цифра броја a која се налази на $i+1$ позицији (гледано са лева на десно), где је i индекс текућег елемента низа. Приказати низ d .
- 1.4 Од низа d направити матрицу $f[5 \times 5]$ на следећи начин. Првих 5 елемената низа d је први ред матрице f , други ред је преосталих 5 елемената. Остале елементе матрице добијамо тако што саберемо елементе из претходна два реда у тој колони. Израчунати суму сваке колоне и ту вредност убацити као елемент главне дијагонале у тој колони. Приказати матрицу.
- 1.5 Ротирати матрицу fz пута на начин као на слици и приказати измењену матрицу.

Оригинална матрица



Матрица након једне ротације

f01	f02	f03	f04	f14
f00	f12	f13	f23	f24
f10	f11	f22	f33	f34
f20	f21	f31	f32	f44
f30	f40	f41	f42	f43

Пример

```
C:\WINDOWS\system32\cmd.exe
```

```
***** ULAZ *****
```

```
Unesite 10-cifreni broj
```

```
1111112223
```

```
Unesite broj rotiranja
```

```
1
```

```
***** IZLAZI *****
```

```
***** odgovor 1 *****
```

```
Srednja vrednost svih cifara je: 1.50
```

```
***** odgovor 2 *****
```

```
Broj 13 je prost broj
```

```
Broj 11 je prost broj
```

```
***** odgovor 3 *****
```

```
0: 1
```

```
1: 6
```

```
2: 3
```

```
3: 1
```

```
4: 1
```

```
5: 1
```

```
6: 2
```

```
7: 2
```

```
8: 2
```

```
9: 3
```

```
***** odgovor 4 *****
```

12	6	3	1	1
1	44	2	2	3
2	8	29	3	4
3	10	7	19	7
5	18	12	8	26

```
***** odgovor 5 *****
```

6	3	1	1	3
12	2	2	3	4
1	44	29	19	7
2	8	10	7	26
3	5	18	12	8

```
Press any key to continue . . .
```

Задатак 2

Унети стринг s величине $4 \times N$ ($2 < N < 10$) карактера који се састоји од малих слова, великих слова енглеске абегеде и цифара. У стрингу не смеју бити две цифре једна иза друге и стринг се не сме завршавати цифром. Претпоставка је да је стринг исправно унет, не треба проверавати да ли је унос исправан. N је број који се не уноси.

- 2.1 Проверити да ли је унети стринг палиндром (занемарити величину слова).
- 2.2 Генерисати нови стринг $s2$ на следећи начин: ако у стрингу s постоји комбинација цифра - велико слово – мало слово, комбинацију треба претворити у цифра + ASCII код великог слова. Ако цифра + ASCII код великог слова прелази 'Z', наставити од 'A'. Приказати $s2$.
- 2.3 Прекопирати стринг s у стринг $s3$. У стрингу $s3$ претворити све цифре у цифра + ASCII код малог слова 'a' и велика слова у мала. Изменити стринг да се састоји само од слова a, b, c и d на следећи начин

ASCII $s3[i]$	$s3[i]$
97-103	a
104-110	b
111-117	c
118-122	d

Израчунати колико најмање измена је потребно да би се у стрингу налазио исти број понављања горе наведених слова. Приказати број измена.

- 2.4 Генерисати нови стринг $s4$ на следећи начин: секвенце подељене цифрама сортирати по тежинском фактору секвенце у растућем поретку. Ако стринг не почиње цифром, додати 0 на почетак стринга.
- 2.5 Прекопирати s у $s5$. Ако стринг $s5$ не почиње цифром, додати 0 на почетак стринга и из стринга $s5$ избацити сва слова која се појављују у свакој секвенци стринга између бројева. Приказати $s5$.

Пример

```
C:\WINDOWS\system32\cmd.exe
```

```
***** ULAZ *****
```

```
Unesite string  
Abcade3Gabcdt9Zaghd1Mtra
```

```
***** IZLAZI *****
```

```
***** odgovor 1 *****
```

```
String nije palindrom
```

```
***** odgovor 2 *****
```

```
AbcadeJbcdtIghdNra
```

```
***** odgovor 3 *****
```

```
Broj izmena: 11
```

```
***** odgovor 4 *****
```

```
0Abcade1Mtra3Gabcdt9Zaghd
```

```
***** odgovor 5 *****
```

```
0Abcde3Gbcdt9Zghd1Mtr
```

```
Press any key to continue . . . █
```

Задатак 3

Улазни подаци су димензије табле и више фигура ($3 < \text{фигуре} < 10$), улазни подаци се раздвајају празним карактером (*space*). Димензије табле су два броја, број колона n и број редова m ($2 < n, m < 10$) и уносе се у формату nm . Фигура је дефинисана симболом (s), позицијом (x, y) и начином кретања (k) и уноси се у формату $sxyk$.

s : 'A' - 'Z'

x : $0 < x \leq n$

y : $0 < y \leq m$

k : { 'X', '/', '\', '+' }

- 'X' – дијагонално кретање у свим правцима,
- '/' – дијагонално кретање у правцу споредне дијагонале и њој паралелим правцима,
- '\' – дијагонално кретање у правцу главне дијагонале и њој паралелним правцима и
- '+' – кретање по хоризонтали и вертикали

Претпоставка је да се почетне позиције фигура не преклапају и да ниједна фигура није „нападнута“.

- 3.1 Исписати таблу као и фигуре које се на њој налазе представљене својим симболима s . На слободним пољима ставити карактер '='. Нумерисати колоне и редове као на примеру.
- 3.2 Исписати таблу, фигуре као и у претходној тачки и додати правце у којима фигуре могу да се крећу. Правац се исписује малим карактерима симбола те фигуре. У случају да се правци секу, на тој позицији ставити карактер '@'.
- 3.3 Испис симбола фигура које прва фигура може да „поједе“.
- 3.4 Број сигурних позиција на које трећа фигура може да се помери. Сигурна позиција је позиција где ниједна друга фигура не може да дође.
- 3.5 Исписати број позиција на којима фигуре могу да се нађу ако могу да направе максимално два помераја (1 померај = 1 поље), без обзира да ли се на тој позицији налази друга фигура. Рачунати и тренутну позицију фигуре.

Пример

```
C:\WINDOWS\system32\cmd.exe
```

```
***** ULAZ *****
```

```
87 A23X B44\ C72+ D47/
```

```
***** IZLAZI *****
```

```
***** odgovor 1 *****
```

```

  1 2 3 4 5 6 7 8
1 = = = = = = =
2 = = = = = C =
3 = A = = = = =
4 = = = B = = =
5 = = = = = = =
6 = = = = = = =
7 = = = D = = =
```

```
***** odgovor 2 *****
```

```

  1 2 3 4 5 6 7 8
1 b = = a = = c =
2 @ @ @ c c c C c
3 = A b = = = c d
4 a = a B = = @ =
5 = = = a b d c =
6 = = = @ b c =
7 = = = D = a @ =
```

```
***** odgovor 3 *****
```

```
D C
```

```
***** odgovor 4 *****
```

```
21
```

```
***** odgovor 5 *****
```

```
A10 B5 C11 D3
```