

Kantonalno takmičenje iz informatike za učenike srednjih škola KS 2019. godine

Sarajevo, 19. mart 2019.

**KANTONALNO TAKMIČENJE IZ INFORMATIKE
ZA UČENIKE SREDNJIH ŠKOLA KANTONA SARAJEVO
ŠKOLSKA 2018/19. GODINA
Sarajevo, 19. 3. 2019.**

KONAČNI REZULTATI - POJEDINAČNO

Pozicija	Prezime i ime	Škola	Bodovi
1.	Imamović Haris	Druga gimnazija Sarajevo	260.00
2.	Ahmespahić Admir	Druga gimnazija Sarajevo	256.88
3.	Dervišević Ishak	Prva bošnjačka gimnazija Sarajevo	212.50
4.	Džaka Tarik	Međunarodna gimnazija Richmond Park Intl.	206.94
5.	Dervišević Adnan	Srednja elektrotehnička škola Sarajevo	200.00
6.	Velašević Boris	Međunarodna gimnazija Richmond Park Intl.	200.00
7.	Šahat Faruk	Druga gimnazija Sarajevo	200.00
8.	Mujkić Amila	Međunarodna gimnazija Richmond Park Intl.	185.77
9.	Grbović Zehra	Prva bošnjačka gimnazija Sarajevo	113.38
10.	Balašev-Šamarski Lav	Druga gimnazija Sarajevo	100.00
11.	Tafro Kanita	Druga gimnazija Sarajevo	95.50
12.	Žgalj Fatih	Prva bošnjačka gimnazija Sarajevo	92.00
13.	Kreho Dženan	Gimnazija Dobrinja	67.00
14.	Vlahovljak Elvir	Treća gimnazija Sarajevo	67.00
15.	Smajević Arslan	Druga gimnazija Sarajevo	55.22
16.	Hamzić Muhamed	Treća gimnazija Sarajevo	50.00
17.	Bajramović Asja	Prva gimnazija Sarajevo	33.00
18.	Guberinić Armin	Srednja elektrotehnička škola Sarajevo	33.00
19.	Sporišević Bilal	Srednja škola za saobraćaj i komunikacije	29.50
20.	Hadžiabdić Kemal	Četvrta gimnazija Ilidža	29.50

Ostali učesnici su osvojili 0 bodova.

KANTONALNO TAKMIČENJE IZ INFORMATIKE
ZA UČENIKE SREDNJIH ŠKOLA KANTONA SARAJEVO
ŠKOLSKA 2018/19. GODINA
Sarajevo, 19. 3. 2019.

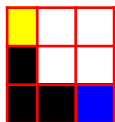
KONAČNI REZULTATI - EKIPNO

Pozicija	Škola	Takmičar	Bodova	Uk. bodova
1.	Druga gimnazija Sarajevo	Ahmespahić Admir	256.88	716.88
		Imamović Haris	260.00	
		Šahat Faruk	200.00	
2.	Međunarodna gimnazija Richmond Park International Secondary School Sarajevo	Džaka Tarik	206.94	592.71
		Velašević Boris	200.00	
		Mujkić Amila	185.77	
3.	Prva bošnjačka gimnazija Sarajevo	Dervišević Ishak	212.50	417.88
		Grbović Zehra	113.38	
		Žgalj Fatih	92.00	
4.	Srednja elektrotehnička škola Sarajevo	Dervišević Adnan	200.00	233.00
		Guberinić Armin	33.00	
5.	Treća gimnazija Sarajevo	Vlahovljak Elvir	67.00	117.00
		Hamzić Muhamed	50.00	
6.	Gimnazija Dobrinja	Kreho Dženan	67.00	67.00
7.	Prva gimnazija Sarajevo	Bajramović Asja	33.00	33.00
8.	Srednja škola za saobraćaj i komunikacije Sarajevo	Sporišević Bilal	29.50	29.50
8.	Četvrta gimnazija Ilidža	Hadžiabdić Kemal	29.50	29.50

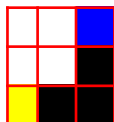
Tetris zmija

Napravite program koji će iscrtavati oblik zmije iz dijelova tetris figura. Ukoliko dođe do sudara dvije figure ili figure i zida, iscrtajte zmiju do tog trenutka (**ne uključujući figuru koja je dovela do sudara**). Figure koje se mogu pojavljivati su:

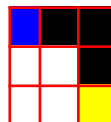
A:



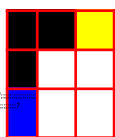
B:



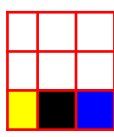
C:



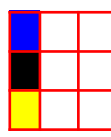
D:



E:



F:



Slika 1

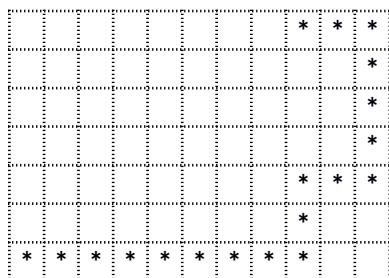
Žuta boja predstavlja početak figure, a plava kraj. Uvijek se nadovezuje sljedeća figura tako da se kraj prethodne figure poveže na početak sljedeće. Prvi početak se postavlja u donji lijevi čošak koordinatnog sistema. Koordinatni sistem ima dvije granice, lijevo i ispod prvog početka.

Ulaz u program je string koji predstavlja figure koje se nadovezuju u redoslijedu kako se pojavljuju u stringu. Na izlazu prikazite izgled figure tako što ćete prikazati koordinatni sistem koji je iscrtan. Visina koordinatnog sistema predstavlja visinu najviše tačke zmije. Širina koordinatnog sistema predstavlja krajnju desnu tačku zmije. Tačke koordinatnog sistema koje zmija zauzima označite zvjezdicama.

Primjer:

EEEEBBC

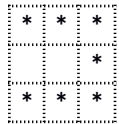
Izlaz:



Primjer:

ECCE

Izlaz:



Format ulaza

U prvom redu se nalazi string koji se sastoji od slova A, B, C, D, E ili F. Pretpostavite da se drugi znakovi neće pojavljivati u stringu.

Format izlaza

Izlaz sadrži određeni broj redova koji se sastoje od znakova razmak ili zvjezdica (asterisk). Broj redova i kolona proizlazi iz postavke zadatka npr. za primjer EEEBBC izlaz će imati 7 redova i 11 kolona jer je to ukupna visina i širina oblika iscrtanog datim figurama u datom redoslijedu.

Ograničenja

Pazite da se program ne izvršava duže od 10s. Maksimalan broj figura je 100.

Rješenje (C++):

```
#include <iostream>
#include <string>

using namespace std;

int pom5X[4][4]={0,0,1,2},{1,2,2,2},{0,0,-1,-2},{-1,-2,-2,-2}};
int pom5Y[4][4]={1,2,2,2},{0,0,-1,-2},{-1,-2,-2,-2},{0,0,1,2}};
int pom3X[2][2]={1,2},{0,0}};
int pom3Y[2][2]={0,0},{-1,-2}};

int Mh=300;
int Mw=0;

bool KoordinatniSistem[301][301];
int x=0;
int y=300;

bool crtaj(char figura){
    int koji = figura-'A';
```

```

//cout<<koji;
if(koji<4){//A-D
    for(int i=0;i<4;i++){
        int xd = pom5X[koji][i];
        int yd = pom5Y[koji][i];
        if(x+xd<0||y+yd>300||KoordinatniSistem[x+xd][y+yd]==true){
            return false;
        }
    }
    for(int i=0;i<4;i++){
        int xd = pom5X[koji][i];
        int yd = pom5Y[koji][i];
        KoordinatniSistem[x+xd][y+yd]=true;
        if(x+xd>Mw)Mw=x+xd;
        if(y+yd<Mh)Mh=y+yd;
    }
    x=x+pom5X[koji][3];
    y=y+pom5Y[koji][3];
    return true;
}else{//E-F
    koji=koji-4;
    for(int i=0;i<2;i++){
        int xd=pom3X[koji][i];
        int yd=pom3Y[koji][i];
        if(x+xd<0||y+yd>300||KoordinatniSistem[x+xd][y+yd]==true){
            return false;
        }
    }
    for(int i=0;i<2;i++){
        int xd=pom3X[koji][i];
        int yd=pom3Y[koji][i];
        //cout<<x+xd<<" "<<y+yd<<endl;
        KoordinatniSistem[x+xd][y+yd]=true;
        if(x+xd>Mw)Mw=x+xd;
        if(y+yd<Mh)Mh=y+yd;
    }
    x=x+pom3X[koji][1];
    y=y+pom3Y[koji][1];
    return true;
}
}

void print(){
    for(int j=Mh;j<=300;j++){
        for(int i=0;i<=Mw;i++){
            cout<<((KoordinatniSistem[i][j])?"*":" ");
        }
        if(j!=300)cout<<endl;
    }
}

int main(){
    for(int i=0;i<=300;i++){
        for(int j=0;j<=300;j++){

```

}

[illegible]

[illegible]

Salvete

Tri drugarice, Anja, Elma i Una, skupljaju salvete. Svaka od njih je skupila određeni broj crvenih, plavih i žutih salveta. Pošto su one najbolje drugarice koje sve dijele, odlučile su da se nađu i razmijene salvete tako da svaka od njih ima podjednak* broj salveta određene boje, kao i podjednak* broj salveta ukupno.

* Pošto ukupan broj salveta ne mora biti djeljiv sa tri, brojevi se smiju razlikovati za najviše 1 salvetu.

Napravite program koji će im pomoći da naprave ovu razmjenu.

Najprije treba unijeti broj salveta za Anju, u vidu tri broja: broj crvenih, plavih i žutih salveta (C P Ž), zatim isto tako za Elmu i za Unu. Program treba zatim ispisati koliko koja djevojčica treba kojoj dati salveta koje boje, u vidu poruka oblika "Anja treba dati Elmi 4 crvenih salveta", svaka poruka u zasebnom redu. Poruke treba ispisati sljedećim redoslijedom: najprije treba ispisati razmjene za crvene salvete, abecednim redom (Anja Elmi, Anja Uni, Elma Anji, Elma Uni, Una Anji, Una Elmi). Ako je broj salveta koje treba razmijeniti 0 onda ne treba ispisati ništa.

Format ulaza

U prvom redu se nalaze tri cijela broja: broj crvenih, plavih i žutih salveta respektivno koje posjeduje Anja.

U drugom redu se nalaze tri broja koji predstavljaju broj salveta za Elmu, te konačno u trećem redu tri broja koji predstavljaju broj salveta za Unu.

Format izlaza

Izlaz treba da sadrži niz tekstualnih poruka tipa:

Anja treba dati Elmi 50 crvenih salveta.

Treba ispisati po jednu poruku za svaku kombinaciju djevojčica koja daje salvete, djevojčica koja dobija salvete i boja salveta, pod uslovom da je broj salveta za datu kombinaciju veći od nule. Redoslijed ispisa treba biti: najprije crvene salvete, zatim plave te na kraju žute, a unutar svake boje poruke trebaju biti poredane abecedno po imenu djevojčice koja dobija salveta, a zatim po imenu djevojčice koja daje salvete.

Primjer 1:

<i>Ulaz:</i>	<i>Izlaz:</i>
150 150 150 0 0 0 150 150 150	Anja treba dati Elmi 50 crvenih salveta. Una treba dati Elmi 50 crvenih salveta. Anja treba dati Elmi 50 plavih salveta. Una treba dati Elmi 50 plavih salveta. Anja treba dati Elmi 50 zutih salveta. Una treba dati Elmi 50 zutih salveta.

Objašnjenje

Ako Anja i Una daju Elmi po 50 salveta svake boje, sve tri djevojčice će imati tačno po 100 salveta svake boje.

Primjer 2:

<i>Ulaz:</i>	<i>Izlaz:</i>
5 4 3 3 4 5 2 2 5	Anja treba dati Uni 1 crvenih salveta. Anja treba dati Uni 1 plavih salveta. Elma treba dati Anji 1 zutih salveta.

Objašnjenje

Finalno stanje nakon razmjene će biti:

$$4 + 3 + 4 = 11$$

$$3 + 4 + 4 = 11$$

$$3 + 3 + 5 = 11$$

Sve tri djevojčice sada imaju jednak ukupan broj salveta (11). Pošto crvenih i plavih salveta ima 10, a žutih 13, ovi brojevi nisu djeljivi sa 3 pa je nemoguće postići da sve tri djevojčice imaju jednak broj, ali se broj svake od ovih boja razlikuje za najviše 1 od jedne do druge djevojčice; Anja ima za 1 više crvenih salveta od Elme i Une, Elma za 1 više plavih od Anje i Une, a Una za 1 više žutih od Anje i Elme.

Ograničenja

Pazite da se program ne izvršava duže od 5s. Nijedan od unesenih brojeva neće biti veći od 1000.

Rješenje (C++):**Testni primjeri:**

Ulaz	Izlaz
5 4 3 3 4 5 2 2 5	Anja treba dati Uni 1 crvenih salveta.\nAnja treba dati Uni 1 plavih salveta.\nElma treba dati Anji 1 zutih salveta.
7 0 0 0 0 0 0 0 0	Anja treba dati Elmi 2 crvenih salveta.\nAnja treba dati Uni 2 crvenih salveta.
7 0 0 0 7 0 0 0 7	Anja treba dati Elmi 2 crvenih salveta.\nAnja treba dati Uni 2 crvenih salveta.\nElma treba dati Anji 2 plavih salveta.\nElma treba dati Uni 2 plavih salveta.\nUna treba dati Anji 2 zutih salveta.\nUna treba dati Elmi 2 zutih salveta.
1 0 0 0 0 0 0 0 0	
7 0 0 0 5 0 0 0 3	Anja treba dati Elmi 2 crvenih salveta.\nAnja treba dati Uni 2 crvenih salveta.\nElma treba dati Anji 1 plavih salveta.\nElma treba dati Uni 2 plavih salveta.\nUna treba dati Anji 1 zutih salveta.\nUna treba dati Elmi 1 zutih salveta.
7 6 0 0 5 0 0 0 3	Anja treba dati Elmi 2 crvenih salveta.\nAnja treba dati Uni 3 crvenih salveta.\nAnja treba dati Uni 2 plavih salveta.\nElma treba dati Uni 1 plavih salveta.\nUna treba dati Elmi 1 zutih salveta.\nUna treba dati Elmi 1 zutih salveta.
2 2 2 0 2 2 0 0 0	Anja treba dati Elmi 1 crvenih salveta.\nAnja treba dati Uni 1 plavih salveta.\nAnja treba dati Uni 1 zutih salveta.\nElma treba dati Uni 1 plavih salveta.\nElma treba dati Uni 1 zutih salveta.
5 5 5 5 0 5 5 5 5	Anja treba dati Elmi 2 plavih salveta.\nUna treba dati Elmi 2 plavih salveta.

Zagrade

U nekom matematičkom izrazu koriste se male (okrugle), srednje (uglaste) i velike (vitičaste) zagrade. Nakon svake otvorene zagrade mora se nalaziti odgovarajuća zatvorena zagrada istog tipa, tako da npr. "[x]" nije legalno jer je otvorena srednja a zatvorena mala zagrada. Također, zagrade različitog tipa se ne smiju ukrštati, tako da ni sljedeći izraz nije legalan: "[(A)]".

Napišite program koji provjerava da li su sve otvorene zagrade ispravno zatvorene. Najprije se unosi broj N koji predstavlja izraza koje treba provjeriti. Zatim se u narednih N redova nalaze stringovi koji predstavljaju matematičke izraze koje treba provjeriti. Ne treba provjeravati da li je izraz kao takav ispravan, samo se provjeravaju zagrade. Program treba za svaki od N stringova ispisati poruku ISPRAVNO ako su sve otvorene zagrade ispravno zatvorene, a NIJE ISPRAVNO ako nisu.

Format ulaza

U prvom redu se nalazi prirodan broj N koji predstavlja broj izraza. U narednih N redova nalaze se stringovi koji predstavljaju matematičke izraze. Stringovi neće sadržavati razmake, a sastojće se isključivo od velikih i malih slova, cifara, zagrada i oznaka matematičkih operacija kao što su + - * / i slično.

Format izlaza

Na izlazu se treba nalaziti N redova teksta od koji svaki red sadrži tekst ISPRAVNO ili NIJE ISPRAVNO.

Primjer

Ulaz:	Izlaz:
(x+y)	ISPRAVNO
[(A)]	NIJE ISPRAVNO
(((1+2+3)))	NIJE ISPRAVNO

Ograničenja

Pazite da se program ne izvršava duže od 10s. N neće imati vrijednost veću od 100.

Rješenje (C++)

```
#include <iostream>
#include <stack>
```

```
using namespace std;
#define NOTOPEN -1
```

```
int main(){
```

```

int n;
cin >> n;
while(n--){
    string s;
    cin >> s;
    stack<char> st;
    bool ok = true;
    for(int i = 0; i < s.length(); i++){
        if(s[i] == '(' || s[i] == '[' || s[i] == '{'){
            st.push(s[i]);
        }else{
            if(s[i] == ')' || s[i] == ']' || s[i] == '}'){
                if(st.empty()){
                    ok = false;
                    goto escape;
                }
                if(s[i] == ')'){
                    if(st.top() != '('){
                        ok = false;
                        break;
                    }else{
                        st.pop();
                    }
                }else if(s[i] == '']){
                    if(st.top() != '['){
                        ok = false;
                        break;
                    }else{
                        st.pop();
                    }
                }else if(s[i] == '}'){
                    if(st.top() != '{'){
                        ok = false;
                        break;
                    }else{
                        st.pop();
                    }
                }
            }
        }
    }
    escape: if(ok && st.empty())
        cout << "ISPRAVNO" << endl;
    else
        cout << "NIJE ISPRAVNO" << endl;
}
}

```

Testni primjeri

Ulaz	Izlaz
------	-------

2 [X) [(A)]	NIJE ISPRAVNO\nNIJE ISPRAVNO
3 (x+y) [(A)] (((1+2+3))	ISPRAVNO\nNIJE ISPRAVNO\nNIJE ISPRAVNO
4))(([(] [(AAA)] (([]	NIJE ISPRAVNO\nNIJE ISPRAVNO\nISPRAVNO\nNIJE ISPRAVNO
1 A	ISPRAVNO
3 {{}} {{()}} [[([x]])]	ISPRAVNO\nISPRAVNO\nISPRAVNO
5 (A)))((((A)))((((A))) (A)))((A)))(((NIJE ISPRAVNO\nNIJE ISPRAVNO\nISPRAVNO\nNIJE ISPRAVNO\nNIJE ISPRAVNO
100 (itd. 100 praznih zagrada	