



| Udruženje nastavnika i profesora informatike | [www.eduit.ba](http://www.eduit.ba) |

**KANTONALNO TAKMIČENJE  
IZ INFORMATIKE ZA UČENIKE OSNOVNIH ŠKOLA 2016**

*Sarajevo, maj 2016. godine*

## KONAČNI REZULTATI KANTONALNOG TAKMIČENJA IZ INFORMATIKE ZA UČENIKE OSNOVNIH ŠKOLA 2016

Mjesto	Učenik	Škola	trojke	xor	recenica	super	UKUPNO BODOVI
1.	Admir Zatega	OŠ Mehmed-beg Kapetanović	110.0	100.0	100.0	100.0	410
2.	Ishak Dervišević	OŠ Srednje	110.0	90.0	50.0	0.0	250
3.	Muhamed Hamzić	OŠ Skender Kulenović	110.0	30.0	100.0	0.0	240
4.	Fatih Žgalj *	OŠ Hamdija Kreševljaković	110.0	0.0	10.0	10.0	130
5.	Emira Ibrahimović *	OŠ Saburina	110.0	0.0	20.0	0.0	130
6.	Tibor Drašković	OŠ Musa Ćazim Ćatić	110.0	0.0	0.0	0.0	110
7.	Zlatan Jakić *	OŠ Hasan Kikić	90.0	0.0	0.0	0.0	90
8.	Azra Šošić *	OŠ 6. Mart	90.0	0.0	0.0	0.0	90
9.	Salahudin Smailagić	OŠ Musa Ćazim Ćatić	70.0	0.0	0.0	0.0	70
10.	Armin Gubernić *	OŠ Skender Kulenović	30.0	30.0	10.0	0.0	70
11.	Muhamed Kalajdžić	OŠ Musa Ćazim Ćatić	60.0	0.0	0.0	0.0	60
12.	Faris Čišija	OŠ Skender Kulenović	50.0	0.0	0.0	0.0	50
13.	Arslan Smajović	OŠ Hilmi-ef. Šarić	40.0	0.0	0.0	0.0	40
14.	Ermin Jaganjac	OŠ Zajko Delić	20.0	0.0	0.0	0.0	20
15.	Dario Telalbašić	OŠ Hasan Kikić	20.0	0.0	0.0	0.0	20
16.	Asja Boračić	OŠ Grbavica I	0.0	0.0	10.0	0.0	10
17.	Elvedin Smajić	Četvrta osnovna škola	0.0	0.0	10.0	0.0	10

Ostali takmičari su ostvarili 0 bodova

\* Učenici su ostvarili jednak broj bodova, ali na osnovu propozicija prednost se daje učeniku koji je ranije završio takmičenje.

## Rezultati po školama

<b>Mjesto</b>	<b>Učenik</b>	<b>Škola</b>	<b>BODOVI</b>	<b>ŠKOLA UKUPNO</b>
<b>1.</b>	Admir Zatega	OŠ Mehmed-beg Kapetanović	<b>410</b>	<b>410</b>
<b>2.</b>	Muhamed Hamzić	OŠ Skender Kulenović	<b>240</b>	<b>360</b>
	Armin Gubernić		<b>70</b>	
	Faris Čišija		<b>50</b>	
<b>3.</b>	Ishak Dervišević	OŠ Srednje	<b>250</b>	<b>250</b>
<b>4.</b>	Tibor Drašković	OŠ Musa Ćazim Ćatić	<b>110</b>	<b>240</b>
	Salahudin Smailbegović		<b>70</b>	
	Muhamed Kalajdžić		<b>60</b>	
<b>5.</b>	Fatih Žgalj *	OŠ Hamdija Kreševljaković	<b>130</b>	<b>130</b>
<b>6.</b>	Emira Ibrahimović *	OŠ Saburina	<b>130</b>	<b>130</b>
<b>7.</b>	Zlatan Jakić *	OŠ Hasan Kikić	<b>90</b>	<b>110</b>
	Dario Telalbašić		<b>20</b>	
<b>8.</b>	Azra Šošić *	OŠ 6. Mart	<b>90</b>	<b>90</b>
<b>9.</b>	Arslan Smajović	OŠ Hilmi-ef. Šarić	<b>40</b>	<b>40</b>
<b>10.</b>	Ermin Jaganjac	OŠ Zajko Delić	<b>20</b>	<b>20</b>
<b>11.</b>	Asja Boračić *	OŠ Grbavica I	<b>10</b>	<b>10</b>
<b>12.</b>	Elvedin Smajić *	Četvrta osnovna škola	<b>10</b>	<b>10</b>

**ZADACI**

**TROJKE**

Napisati program koji omogućuje unos **neograničenog** niza **pozitivnih** cijelih brojeva sa tastature, a zatim ispisuje broj članova niza koji su dvostruko veći od prethodnog člana, a dvostruko manji od sljedećeg člana niza. Naravno, prvi i posljednji član niza automatski ne zadovoljavaju uslov zadatka.

**Format ulaza**

Unosi se **neograničen** broj pozitivnih cijelih brojeva u zasebnim redovima, a unos se završava brojem nula (koji nije pozitivan pa samim time nije član niza). U testnim primjerima može biti jako veliki broj brojeva na ulazu, ali vrijeme predviđeno za test će biti dovoljno tako da ne morate razmišljati o vremenu izvršenja.

Pretpostavite da korisnik neće unositi negativne brojeve te da će unijeti najmanje tri člana niza (ne računajući nulu kojom se završava unos).

**Format izlaza**

Program ispisuje jedan cijeli broj. Ovaj broj neće biti veći od maksimalne cjelobrojne vrijednosti koju podržavaju programski jezici dozvoljeni na takmičenju.

**Primjer**

Radi kraćeg zapisa u primjeru ispod su brojevi razdvojeni razmakom, ali u stvarnom programu svaki broj na ulazu treba biti u zasebnom redu.

<i>Ulaz:</i>	<i>Izlaz:</i>
2 4 8 16 33 66 132 0	3

Članovi niza koji zadovoljavaju uslov zadatka su: 4, 8, 66 (ukupno tri člana niza).

**XOR**

Dat je neki niz prirodnih brojeva. Ako na svaki član tog niza primijenimo operaciju binarni XOR (ekskluzivna disjunkcija) sa nekim prirodnim brojem X dobićemo neki drugi niz prirodnih brojeva.

Recimo da niz glasi 4 2 3 1. Ako je  $X=6$  dobićemo niz 2 4 5 6. Objašnjenje

Decimalni zapis	4	2	3	1
Binarni zapis	100	010	011	001
Broj $X=6$	110	110	110	110
Nakon XOR-a	010	100	101	111
Decimalni zapis	2	4	5	6

Vaš zadatak je da napravite program koji za uneseni niz provjerava da li postoji X takvu da se nakon primjene XOR dobije niz koji je sortiran u rastućem poretku (ali ne strogo rastućem, dakle dozvoljeno je da dva susjedna člana budu jednaka). U prethodnom primjeru za niz 4 2 3 1 postoji takvo X i ono iznosi 6.

Program treba ispisati X ili, ako X ne postoji, treba ispisati -1.

**Format ulaza**

U prvom redu se nalazi prirodan broj  $N \leq 10$  koji predstavlja broj članova niza. Nakon toga se unosi N brojeva razdvojenih razmacima.

**Format izlaza**

Program ispisuje jedan cijeli broj X čije je značenje objašnjeno u primjeru ili -1 ako takvo X ne postoji.

**Primjer**

<i>Ulaz:</i>	<i>Izlaz:</i>
4 4 2 3 1	6

**REČENICA**

Na ulazu programa se nalazi neka rečenica dužine najviše 2000 karaktera. Vaš zadatak je da ispišete na ekran unesenu rečenicu transformisanu na sljedeći način:

- Malo početno slovo svake riječi treba zamijeniti odgovarajućim velikim slovom.
- Riječ u kojoj se pojavljuje najveći broj velikih ili malih slova A treba zamijeniti riječju OVDJE. Ako se u čitavoj rečenici ne pojavljuje niti jedno slovo A, ne treba raditi zamjenu. Ako se pojavljuje nekoliko riječi sa istim (najvećim) brojem slova A treba zamijeniti prvu takvu riječ gledano s lijeva.
- Svaka riječ duža od devet slova zamijenjena je sa prva tri slova i znakom tačka (npr. riječ "skracenica" treba ispisati kao "skr.")

Pretpostavlja se da je riječ neki niz znakova koji je sa obje strane omeđen znakom razmaka, početkom ili krajem stringa. Ne trebate provjeravati da li se riječi sastoje iz slova ili sadrže i neke znakove interpunkcije, trebate jednostavno pretpostaviti da su svi znakovi slova, a znakove interpunkcije računati u riječ.

**Format ulaza**

Tražena rečenica se čita sa standardnog ulaza (tastature) ILI iz datoteke pod imenom "input.txt" koja se nalazi u tekućem direktoriju (bez navođenja puta). Oba rješenja će biti priznata. Unos se završava znakom ENTER (karakterom za prelazak u novi red).

**Format izlaza**

Transformisanu rečenicu treba ispisati na ekran.

**Primjer**

Radi kraćeg zapisa u primjeru ispod rečenica je razdvojena na redove, ali u stvarnom programu ulaz i izlaz bi trebali biti u

<i>Ulaz:</i>	<i>Izlaz:</i>
Dobro dosli na kantonalno takmicenje iz informatike u Sarajevu.	Dobro Dosli Na OVDJE Tak. Iz Inf. U Sarajevu.

## SUPER SLOŽENI BROJEVI

Poznato vam je da pored dekadskog postoje i drugi *brojevnih sistemi*, pa tako npr. imamo binarni brojevni sistem koji posjeduje samo dvije cifre: 0 i 1, zatim ternarni sistem čije su cifre 0,1,2 itd. Kažemo da dekadski sistem ima bazu 10, binarni ima bazu 2, ternarni bazu 3 itd.

Dalje vam je poznato da za prirodan broj veći od 1 kažemo da je *prost broj* ako nije djeljiv ni sa jednim brojem osim sa 1 i sa samim sobom, a ako nije prost onda se naziva *složenim brojem*. Ova dva pojma ćemo povezati i uvesti pojam *super-složenog broja*. To je broj koji je složen ne samo u dekadskom brojevnom sistemu, nego i u svim brojevnim sistemima čije su baze manje od 10 a veće od 1 – naravno, ako ove brojeve čitamo kao da su u dekadskom sistemu.

Recimo broj 10 u dekadskom sistemu (pišemo ga kao  $10_{10}$ ) u binarnom sistemu ima vrijednost  $1010_2$ , u ternarnom  $101_3$  itd. Broj 10 je složen u binarnom sistemu ( $1010$  je djeljivo sa dva), no u ternarnom nije jer broj 101 kada ga čitamo kao u dekadskom sistemu je prost broj, tako da broj 10 nije super-složen broj.

Vaš zadatak je da napišete program koji provjerava da li su brojevi super-složeni.

### Format ulaza

U prvom redu se nalazi prirodan broj  $N \leq 10$  koji predstavlja broj članova niza. Nakon toga se unosi N brojeva razdvojenih razmacima, pri čemu niti jedan član niza nije veći od 10000.

### Format izlaza

Program ispisuje sve one članove niza koji su supersloženi brojevi. Ako niti jedan član niza nije supersložen ispisuje se 0 (koja nije ni prost ni složen broj).

### Primjer

Ulaz:	Izlaz:
5	51 69 99 125 129
51 69 99 125 129	

Svi navedeni brojevi su super-složeni. Npr. broj  $99_{10}$  u drugim brojevnim sistemima je  $1100011_2$   $10200_3$   $1203_4$   $344_5$   $243_6$   $201_7$   $143_8$  i  $120_9$ . Svi ovi brojevi su složeni brojevi u dekadskom sistemu.



## **PRIJEDLOG MOGUĆIH RJEŠENJA**

## TROJKE – C++

```
1  #include <iostream>
2
3  using namespace std;
4
5  int main() {
6      int br1, br2, br3, br(0);
7      cin >> br1 >> br2 >> br3;
8      do {
9          if (br3 == br2*2 && br3 == br1*4)
10             br++;
11             br1=br2;
12             br2=br3;
13             cin >> br3;
14         } while (br3 != 0);
15         cout << br;
16         return 0;
17     }
```

## XOR – C++

```

1  #include <iostream>
2  #include <vector>
3  using namespace std;
4  int nbb(int a, int b)
5  {
6      unsigned t=1;
7      while (t<=a || t<=b) {
8          t<<=1;
9      }
10     return t/2;
11 }
12 int main()
13 {
14     int n,m,c;
15
16     cin>>n;
17     vector<int> xi;
18     for(int i=0;i<n;i++){
19         cin>>c;
20         xi.push_back(c);
21     }
22     int zadrziMaska=0;
23     int zamjeniMaska=0;
24     for(int i=0;i<n;i++){
25         for(int j=i+1;j<n;j++){
26             if(xi[i]>xi[j]) zamjeniMaska=zamjeniMaska | nbb(xi[i]^xi[j],0);
27             else if(xi[i]<xi[j]) zadrziMaska=zadrziMaska | nbb(xi[i]^xi[j],0);
28         }
29     }
30     if(nbb(zamjeniMaska&zadrziMaska,0)==0 || zamjeniMaska==0) cout<<zamjeniMaska;
31     else cout<<-1;
32     return 0;
33 }

```

## REČENICA – C++

```

1  #include <string>
2  #include <iostream>
3
4  using namespace std;
5
6  int main() {
7      int brslova(0), bra(0), maxa(0), maxapos(-1);
8      string s;
9      getline(cin, s);
10
11
12     for (int i(0); i<=s.length(); i++) {
13         if (i==s.length() || s[i] == ' ') {
14             if (bra > maxa) {
15                 maxa = bra;
16                 maxapos = i;
17             }
18             bra = 0;
19         } else if (s[i] == 'a' || s[i] == 'A') bra++;
20     }
21
22
23     for (int i(0); i<=s.length(); i++) {
24         if (i==s.length() || s[i] == ' ') {
25             if (i == maxapos) cout << "OVDJE ";
26             else {
27                 if (s[i-brslova] >= 'a' && s[i-brslova] <= 'z')
28                     s[i-brslova] -= 32;
29                 if (brslova > 9) {
30                     cout << s[i-brslova] << s[i-brslova+1] << s[i-brslova+2] << ". ";
31                 } else {
32                     for (int j(i-brslova); j<=i && j<s.length(); j++)
33                         cout << s[j];
34                 }
35             }
36             brslova=0;
37         } else {
38             brslova++;
39         }
40     }
41
42     cout << "\n";
43
44     return 0;
45

```

## SUPERSLOŽENI – C++

```

1  #include <iostream>
2  #include <climits>
3  #include <vector>
4  #include <cmath>
5
6  using namespace std;
7
8  bool prost(long long x) {
9      int korijen=sqrt(x);
10     if (x%2 == 0) return false;
11     for (int i(3); i<=korijen; i+=2)
12         if (x%i == 0) return false;
13     return true;
14 }
15
16 long long konverzija(long long x, int baza) {
17     long long rez=0;
18     long long tk=1;
19     while (x>0) {
20         rez = rez+(x%baza)*tk;
21
22         tk = tk*10;
23         x/=baza;
24     }
25     return rez;
26 }
27
28 bool superslozen(long long x) {
29     long long broj;
30     for (int baza(2); baza<=10; baza++) {
31         if (baza==10) broj=x; else broj=konverzija(x,baza);
32
33         if (prost(broj)) return false;
34     }
35     return true;
36 }
37
38 int main() {
39     int N, x;
40     cin >> N;
41     for (int i(0); i<N; i++) {
42         cin >> x;
43         if (superslozen(x)) {
44             cout << x << " ";
45         }
46     }
47     return 0;
48 }

```