

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

Sarajevo, 25. mart 2019.

**KANTONALNO TAKMIČENJE IZ INFORMATIKE
ZA UČENIKE OSNOVNIH ŠKOLA KANTONA SARAJEVO
ŠKOLSKA 2018/19. GODINA
Sarajevo, 25. 3. 2019.**

KONAČNI REZULTATI - POJEDINAČNO

| Pozicija | Prezime i ime | Škola | Bodovi |
|----------|-------------------|------------------------------------|--------|
| 1. | Ibrahimović Emira | OŠ Saburina | 290 |
| 1. | Zatega Admir | OŠ Mehmed-beg Kapetanović Ljubušak | 290 |
| 2. | Bećirović Muhamed | OŠ Zahid Baručija | 270 |
| 3. | Fetahović Tarik | OŠ Grbavica II | 190 |
| 4. | Čišija Faris | OŠ Skender Kulenović | 180 |
| 5. | Bašić Asmin | OŠ Aleksa Šantić | 160 |
| 6. | Ibrahimović Faruk | OŠ Saburina | 130 |
| 7. | Alibegović Harun | OŠ Osma osnovna škola | 120 |
| 8. | Pavlinović Ema | OŠ Osman Nakaš | 100 |
| 9. | Džafo Faris | OŠ Hašim Spahić | 90 |
| 10. | Japalak Bakir | OŠ Hilmi ef. Šarić | 60 |
| 11. | Cvrk Ilma | OŠ Izet Šabić | 50 |
| 12. | Gajić Luka | OŠ Hasan Kikić | 50 |
| 13. | Jakupović Esma | OŠ Kovačići | 40 |
| 14. | Šošić Azra | OŠ Hadžići | 30 |
| 15. | Galijašević Tarik | OŠ Skender Kulenović | 30 |
| 16. | Ikonić Mak | OŠ Isak Samokovlija | 20 |
| 17. | Karić Kamran | OŠ Musa Ćazim Ćatić | 10 |

Ostali učesnici su osvojili 0 bodova.

KANTONALNO TAKMIČENJE IZ INFORMATIKE
ZA UČENIKE OSNOVNIH ŠKOLA KANTONA SARAJEVO
ŠKOLSKA 2018/19. GODINA
Sarajevo, 25. 3. 2019.

KONAČNI REZULTATI - Škole

| Pozicija | Škola | Takmičar | Bodova | Uk. bodova |
|----------|------------------------------------|-------------------|--------|------------|
| 1. | OŠ Saburina | Ibrahimović Emira | 290 | 420 |
| | | Ibrahimović Faruk | 130 | |
| 2. | OŠ Mehmed-beg Kapetanović Ljubušak | Zatega Admir | 290 | 290 |
| 3. | OŠ Zahid Baručija | Bećirović Muhamed | 270 | 270 |
| 4. | OŠ Skender Kulenović | Čišija Faris | 180 | 210 |
| | | Galijašević Tarik | 30 | |
| 5. | OŠ Grbavica II | Fetahović Tarik | 190 | 190 |
| 6. | OŠ Aleksa Šantić | Bašić Asmin | 160 | 160 |
| 7. | OŠ Osma osnovna škola | Alibegović Harun | 120 | 120 |
| 8. | OŠ Osman Nakaš | Pavlinović Ema | 100 | 100 |
| 9. | OŠ Hašim Spahić | Džafo Faris | 90 | 90 |
| 10. | OŠ Hilmi ef. Šarić | Japalak Bakir | 60 | 60 |
| 11. | OŠ Izet Šabić | Cvrk Ilma | 50 | 50 |
| 11. | OŠ Hasan Kikić | Gajić Luka | 50 | 50 |
| 13. | OŠ Kovačići | Jakupović Esmā | 40 | 40 |
| 14. | OŠ Hadžići | Šošić Azra | 30 | 30 |
| 15. | OŠ Isak Samokovlija | Ikonić Mak | 20 | 20 |
| 16. | OŠ Musa Ćazim Ćatić | Karić Kamran | 10 | 10 |

Ponavljanje slova

Kada napišete jednu riječ, a onda napišete istu riječ ispod prve pomaknutu ulijevo za jedno slovo. Isti proces ponovite, svaki put pomjerajući za jedno slovo ulijevo riječ koju pišete, sve dok na papiru ne bude onoliko riječi koliko ima slova u prvoj riječi. Tada možete izbrojati koliko ima jednakih slova na istim pozicijama u početnoj riječi i u svakoj potpisanoj riječi. Ovaj broj slova ćemo nazvati faktor ponavljanja. Vaš zadatak je da izračunate faktor ponavljanja za bilo koju unesenu riječ.

Primjer 1.)

```

BANANA
BANANA
BANANA
BANANA
BANANA
BANANA

```

F=4

Primjer 2.)

```

JABUKA
JABUKA
JABUKA
JABUKA
JABUKA
JABUKA

```

F=1

Ulaz:

Unosi se jedna riječ napisana velikim slovima engleske abecede. Riječ neće biti duža od 1000 znakova.

Izlaz:

Na izlazu u jednoj liniji trebate ispisati $F=X$, gdje X predstavlja faktor ponavljanja za unesenu riječ.

Primjer:

JABUKA

Izlaz:

F=1

Format ulaza

U prvom redu se nalazi string koji predstavlja jednu riječ.

Format izlaza

Izlaz sadrži jednu liniju oblika $F=X$ gdje je X predstavlja faktor ponavljanja za datu riječ.

Ograničenja

Pazite da se program ne izvršava duže od 10s.

Rješenje (C++):

```
#include <iostream>

using namespace std;

int main(){
    string s;
    cin>>s;
    int ponavljanje=0;
    for(int pomak=1;pomak<s.length();pomak++){
        for(int i=0;i<s.length();i++){
            if(s[i]==s[i+pomak])ponavljanje++;
        }
    }
    cout<<"F="<<ponavljanje;
}
```

Rješenje (Qbasic):

```
CLS
INPUT a$
FOR i = 1 TO LEN(a$)
    FOR j = 1 TO LEN(a$)
        IF MID$(a$, i, 1) = MID$(a$, i + j, 1) THEN r = r + 1
    NEXT j
NEXT i
PRINT "F="; r
END
```

Testni primjeri:

| Ulaz | Izlaz |
|-------------------------------|-------|
| BANANA | F=4 |
| RADARRADARRADARRADARRADARRADA | |

| | |
|---------------------|-------|
| RRADARRADARRADAR | F=425 |
| ANANASANANAS | F=22 |
| AAAAAAAAAA | F=45 |
| TRATARATIRA | F=12 |
| KUSKUS | F=3 |
| STOLICA | F=0 |
| PUTOPIS | F=1 |
| EVOSADASOVE | F=5 |
| AMENETUNIMINUTENEMA | F=19 |
| | |

Suma N u M

Poznato vam je da postoji formula za zbir svih prirodnih brojeva do broja N. Ta formula glasi $S = N * (N + 1) / 2$. Ova formula vrijedi za sve prirodne brojeve u dekadnom sistemu. Vaš zadatak je da sračunate koja je vrijednost sume brojeva od 1 do broja N, ali tako što prvo brojeve zapišete u nekom drugom brojnom sistemu pa ih tek onda saberete. Napišite program koji će sračunati zbir svih brojeva do broja N kada su zapisani u brojnom sistemu M. Broj M je prirodan broj i uvijek je manji ili jednak 10, a N je prirodan broj zapisan u dekadnom sistemu.

Format ulaza

Unose se dva broja. U prvom redu je broj N i on predstavlja broj u dekadnom sistemu do kojeg treba sračunati sumu. U drugom redu je broj M, on predstavlja brojni sistem u kojeg trebate brojeve pretvoriti prije sabiranja. Broj N je uvijek veći od 0 i manji od 10000. Broj M može biti od 1 do 10.

Format izlaza

Na izlazu se treba nalaziti samo jedan broj koji predstavlja sračunatu sumu.

Primjer 1:

| Ulaz: | Izlaz: |
|---------|--------|
| 10 3 | 300 |

Objašnjenje

U ovom slučaju $N=10$, a $M=3$
Prvih 10 brojeva u brojnom sistemu 3 je:
1,2,10,11,12,20,21,22,100,101
Suma ovih brojeva je 300

Primjer 2:

| Ulaz: | Izlaz: |
|--------|--------|
| 5 5 | 20 |

Objašnjenje

$N=5, M=5$

Prvih 5 brojeva u brojnom sistemu 5 je:
1,2,3,4,10
Suma ovih brojeva u decimalnom sistemu je 20

Ograničenja

Pazite da se program ne izvršava duže od 1-s. Broj N je uvijek veći od 0 i manji od 10000. Broj M može biti od 1 do 10.

Rješenje (C++):

```
#include <iostream>

using namespace std;

int dajSumuNuM(int b, int n){
    long long nn=1;
    long long nd=1;
    long long rez=0;
    while(b/nn!=0){
        for(long long b0=nn-1, val=0; b0<=b; b0+=nn, val=(val+1)%n) {
            rez+=nd*nn*val;
            if(b0+nn>b){
                rez+=nd*(b-b0)*((val+1)%n);}
        }
        nd*=10;
        nn*=n;
    }
    return rez;
}

int main(){
    int n,m;
    cin>>n;
    cin>>m;
    cout<<dajSumuNuM(n,m);
}
```

Testni primjeri:

| Ulaz | Izlaz |
|----------|-------|
| 5 5 | 20 |
| 100 5 | 17600 |
| 5 1 | 12345 |
| 500 | |

| | |
|------------|------------|
| 2 | 1452421874 |
| 500 8 | 190426 |
| 7 8 | 28 |
| 7 10 | 28 |
| 1000 10 | 500500 |
| 1000 7 | 1297513 |
| 1000 5 | 5610000 |
| | |

Dvostruko

Potrebno je napisati program koji omogućuje unos nekog teksta, a zatim sve brojeve u tom tekstu množi sa dva, te na ekranu ispisuje tekst u kojem su brojevi pomnoženi sa dva.

Npr. ako korisnik unese:

U ovoj recenici se nalazi broj 4, broj 8 i broj 88

program treba ispisati:

U ovoj recenici se nalazi broj 8, broj 16 i broj 176

Format ulaza

Na ulazu se nalazi jedna linija teksta koji predstavlja rečenicu. Rečenica se sastoji od slova i brojeva. Pod pojmom "broj" podrazumijeva se bilo koji neprekinuti niz cifara 0-9. Pretpostavite da su svi brojevi cijeli. Primijetite da će funkcija raditi ispravno i sa negativnim brojevima a ko samo zanemarite minus.

Format izlaza

Na izlazu se treba nalaziti jedna linija teksta koja je identična unesenoj, pri čemu su svi brojevi pomnoženi sa dva.

Ograničenja

Pazite da se program ne izvršava duže od 10s. Nijedan broj neće biti veći od 100000.

Rješenje (C++):

```
#include <iostream>
#include <sstream>
#include <cstdlib>
#include <cstring>

using namespace std;

int main()
{
    string recenica;
    getline(cin, recenica);

    string broj = "";
    for(int i = 0; i < recenica.length(); i++){
        if(recenica[i] < '0' || recenica[i] > '9'){
            cout << recenica[i];
            continue;
        }
    }
```

```

    broj = "";
    while(recenica[i] >= '0' && recenica[i] <= '9' && i < recenica.leng
th()){
        broj += recenica[i];
        i++;
    }

    int broji = atoi(broj.c_str());
    broji *= 2;
    stringstream cnv;
    cnv << broji;
    cout << cnv.str();

    if(i != recenica.length())
        i--;
}

cout << endl;

return 0;
}

```

Rješenje (QBasic):

```

DIM rec AS STRING
DIM nrec AS STRING
INPUT rec
FOR x = 1 TO LEN(rec)
    t$ = MID$(rec, x, 1)
    IF t$ >= "0" AND t$ <= "9" THEN
        b$ = b$ + t$
    ELSE s = VAL(b$) * 2: b$ = ""
        IF s <> 0 THEN nrec = nrec + LTRIM$(STR$(s))
        nrec = nrec + t$
    END IF
NEXT x
IF b$ <> "" THEN s = VAL(b$) * 2: nrec = nrec + LTRIM$(STR$(s))

PRINT nrec

```

Testni primjeri:

| Ulaz | Izlaz |
|--|--|
| U ovoj recenici se nalazi broj 4, broj 8 i broj 88 | U ovoj recenici se nalazi broj 8, broj 16 i broj 176 |
| 1 i 1 jesu 2 | 2 i 2 jesu 4 |

| | |
|--|---|
| 76543 | 153086 |
| Moze li broj 999999 | Moze li broj 1999998 |
| 5 5 5 5 5 5 5 5 5 5 5 5 5 5 | 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 |
| Radi li za broj 0 i -1? | Radi li za broj 0 i -2? |
| 53719 | 107438 |
| aaa234aaa | aaa468aaa |
| aaa579aaa | aaa1158aaa |
| Za broj 044 ispravno rjesenje je i 088 i 88. | Za broj 088 ispravno rjesenje je i 0176 i 176. |