

PRIRODNO-MATEMATIČKI FAKULTET
UNIVERZITET U SARAJEVU

Zbirka zadataka sa riješenim zadacima – Programski jezik C++

Prvi dio

Sarajevo
2021.

1. Napišite C++ program koji prima podatke o dva vremenska trenutka, izražena u satima, minutama i sekundama, te izračunava i prikazuje proteklo vrijeme između ta dva trenutka, također izraženo u satima, minutama i sekundama. Pretpostavite da je drugi vremenski trenutak uvijek kasniji od prvog. Na primjer, prilikom pokretanja programa, korisnik će biti zamoljen da unese informacije o oba vremenska trenutka, kao ispod.:

Unesi prvi vremenski trenutak (h m s): 15 15 7

Unesi drugi vremenski trenutak (h m s): 17 9 34

Između dva vremenska trenutka prošlo je 1h 54min 27s.

Rješenje.

```
#include<iostream>
int main() {
    int h1,m1,s1,h2,m2,s2;
    std::cout<<"Unesite prvi vremenski trenutak (h,min,s): ";
    std::cin>>h1>>m1>>s1;
    std::cout<<"Unesite drugi vremenski trenutak (h,min,s): ";
    std::cin>>h2>>m2>>s2;
    int suma_sekundi(3600*(h2-h1)+60*(m2-m1)+s2-s1);
    std::cout<<"Izmedju dva vremenska trenutka proslo je "<<
suma_sekundi/3600<<"h "<<((suma_sekundi-suma_sekundi%60)/60)%60<<"min
"<<suma_sekundi%60<<"s.";
    return 0;
}
```

2. Napišite C++ program koji traži unos brzine broda u čvorovima preko tastature kao cijeli broj. Zatim izračunajte i prikazite brzinu broda u km/h kao decimalni broj. Za pretvaranje koristite činjenicu da je jedan čvor jednak jednoj morskoj milji na sat, a jedna morska milja ekvivalentna 1852 metara (definirana kao konstanta). Na primjer, ako korisnik unese brzinu broda kao 20 čvorova, program bi trebao prikazati rezultat kao 37.04 km/h, jer je 20 čvorova ekvivalentno 37.04 km/h.

Rješenje.

```
#include<iostream>
int main() {
    int brzina_broda_u_cvorovima;
    const int morska_milja(1852);
    std::cout<<"Unesite brzinu broda u cvorovima: ";
    std::cin>>brzina_broda_u_cvorovima;
    double brzina(brzina_broda_u_cvorovima*morska_milja/1000.0);
    std::cout<<"Brzina u km/h je: "<<brzina;
    return 0;
}
```

3. Napišite C++ program koji zahtijeva unos podatka sa tastature, a zatim ispisuje jedan od sljedećih pet komentara u zavisnosti od unijetog podatka:

Uneseni podatak je prirodan broj.

Uneseni podatak je cijeli broj, ali nije prirodan.

Uneseni podatak je realan broj, ali nije cijeli.

Uneseni podatak nije broj.

Rješenje.

```
#include<iostream>
int main() {
    double x;
    std::cout<<"Unesite broj: " ;
    std::cin>>x;
    if(!std::cin) std::cout<<"Uneseni podatak nije broj!";
    else if((x-int(x))==0) {
        if(x>0) std::cout<<"Uneseni broj je prirodan.";
        else std::cout<<"Uneseni broj je cijeli, ali nije prirodan.";
    }
    else std::cout<<"Uneseni broj je realan, ali nije cijeli.";
    return 0;
}
```

4. Napišite C++ program koji od korisnika traži da unese željeni broj sati korištenja interneta, a zatim preporučuje koja je od dvije firme isplativija za njega. Prva firma naplaćuje fiksnu pretplatu od 10 KM plus 50 feninga po svakom potrošenom satu, dok druga firma ne naplaćuje fiksnu pretplatu, već traži 80 feninga po svakom potrošenom satu. Na primjer, za 15 sati korištenja, isplativija je druga firma (jer je $10+15\cdot0.5=17.5 > 15\cdot0.8=12$), dok je za 40 sati isplativija prva firma (jer je $10+40\cdot0.5=30 < 40\cdot0.8=32$).

Rješenje.

```
#include<iostream>
int main() {
    double broj_sati;
    std::cout<<"Unesite broj sati: ";
    std::cin>>broj_sati;
    double prva_firma(10+0.5*broj_sati), druga_firma(0.8*broj_sati);
    if(prva_firma>druga_firma) std::cout<<"Druga firma je isplativija!";
    else std::cout<<"Prva firma je isplativija!";
    return 0;
}
```

5. Napišite C++ program koji traži unos koeficijenata a , b i c za kvadratnu jednačinu $ax^2 + bx + c = 0$. Program treba izračunati i ispisati rješenja te

jednačine. Osigurajte da se pri unosu svakog koeficijenta uvijek čitaju svježi podaci, bez obzira na to što korisnik može odmah unijeti sva tri podatka prilikom unosa prvog koeficijenta. Program treba rukovati svim specijalnim slučajevima, uključujući situacije kada je $a = 0$ (što jednačinu svodi na linearnu, s jednim rješenjem ako je $b \neq 0$), kada su $a = b = 0$ i $c \neq 0$ (kada nema rješenja), te kada su $a = b = c = 0$ (kada je jednačina identički zadovoljena za sve vrijednosti x). Također, treba predvidjeti mogućnost kompleksnih rješenja, koja će biti ispisana kao uređeni parovi realnih brojeva.

Rješenje.

```
#include<iostream>
#include<cmath>
void ocisti() {std::cin.clear();std::cin.ignore(1000, '\n');} //pom.funkcija
int main() {
    double a,b,c;
    std::cout<<"Unesite koeficient uz x^2: "; std::cin>>a; ocisti();
    std::cout<<"Unesite koeficient uz x: "; std::cin>>b; ocisti();
    std::cout<<"Unesite slobodni koeficient: "; std::cin>>c; ocisti();
    if(a==0) {
        if(b==0 && c!=0) std::cout<<"Jednacina nema rjesenja!";
        else if(b==0 && c==0) std::cout<<"Jednacina ima besk.mnogo rjesenja!";
        else std::cout<<"x= "<<-c/b ;
    }
    else{
        double D(b*b-4*a*c);
        if(D==0) std::cout<<"x1 = x2 = "<<-b/2*a;
        else if (D>0) {
            std::cout<<"x1 = " <<(-b+sqrt(D))/2*a<<std::endl;
            std::cout<<"x2 = " <<(-b-sqrt(D))/2*a<<std::endl;
        }
        else{
            if(a>0) {
                std::cout<<"x1="<<-b/2*a<<" + "<<sqrt(-D)/2*a<<"i"<<std::endl;
                std::cout<<"x2=" <<-b/2*a<<" - "<<sqrt(-D)/2*a<<"i"<<std::endl;
            }
            else{
                std::cout<<"x1="<<-b/2*a<<" - "<<sqrt(-D)/2*a<<"i"<<std::endl;
                std::cout<<"x2="<<-b/2*a<<" + "<<sqrt(-D)/2*a<<"i"<<std::endl;
            }
        }
    }
    return 0;
}
```

6. Napišite C++ program koji od korisnika traži unos tri realna broja a , b i c , a zatim ispisuje da li te tri vrijednosti mogu predstavljati stranice pravouglog trougla. Program bi trebao prepoznati kombinacije brojeva poput 3, 4, 5 ili 5, 4, 3 ili 3, 5, 4 ili 0.3, 0.4 i 0.5 kao stranice pravouglog trougla.

Rješenje.

```
#include<iostream>
#include<cmath>
int main() {
    double a,b,c;
    std::cout<<"Unesite stranice trouglova: ";
    std::cin>>a>>b>>c;
    if(a>b){if(a>c){double temp(c);c=a;a=temp;}}
    else if(b>c){double temp(c);c=b;b=temp;}
    if(a+b<=c||b+c<=a||c+a<=b)std::cout<<"Stranice ne formiraju prav.trokut.";
    else if(fabs(c*c-a*a-b*b)<1e-10)std::cout<<"Stranice formiraju prav. tr.";
    else std::cout<<"Stranice ne formiraju pravougli trougao.";
    return 0;
}
```

7. Napišite C++ program koji traži unos glavnice G i kamatne stope. Program treba izračunati iznos kamate K koristeći formulu $K = G * (stopa/100)$, te novu stopu $S = G + K$. Nakon toga, program treba ispisati sve unesene i izračunate podatke u sređenom formatu, s poravnanjem svih brojeva uz desnu ivicu, pri čemu će sve cifre jedinica biti poravnate jedna ispod druge. Na primjer, za unos $G = 1000$ i $stopa = 5$, ispis će biti:

Glavnica: 1000

Kamatna stopa: 5

Kamate: 50

Nova svota: 1050

a ako se unese $G = 55000$ i $stopa = 12$, dobija se:

Glavnica: 55000

Kamatna stopa: 12

Kamate: 6600

Nova svota: 61600

Pretpostavite da nijedna veličina neće preći širinu od 6 cifara. Eventualne decimalne rezultate zaokružite, tako da za $G = 3530$ i $stopa = 7$, kamate trebaju biti $K = 247$ a ne 247.1.

Rješenje.

```
#include<iostream>
#include<iomanip>
int main() {
    unsigned int Glavnica;
    int kamatna_stopu;
    std::cout<<"Unesite glavnica: ";
    std::cin>>Glavnica;
    std::cout<<"Unesite kamatnu stopu: ";
    std::cin>>kamatna_stopu;
    int Kamata(Glavnica*kamatna_stopu/100);
    int Nova_Svota(Glavnica+Kamata);
```

```
std::cout<<"Glavnica :      "<<std::setw(6)<<Glavnica<<std::endl;
std::cout<<"Kamatna stopa : "<<std::setw(6)<<kamatna_stopa<<std::endl;
std::cout<<"Kamate :      "<<std::setw(6)<<Kamata<<std::endl;
std::cout<<"Nova svota :    "<<std::setw(6)<<Nova_Svota<< std::endl;
return 0;
}
```

8. Napišite C++ program koji traži unos 6 brojeva sa tastature, a zatim ispisuje koliko unesenih brojeva su pozitivni, i koliko među njima ima neparnih brojeva. Program ne smije koristiti nizove ili druge srodne strukture podataka. Testirajte program na sljedećim primjerima:

Primjer 1: 2 6 10 18 8 6

Primjer 2: 4 -12 10 18 -18 10

Primjer 3: 3 5 4 19 7 12

Primjer 4: -3 9 15 -7 13 11

Rješenje.

```
#include<iostream>
int main() {
    int a,b,c,d,e,f;
    std::cout<<"Unesite 6 brojeva: ";
    std::cin>>a>>b>>c>>d>>e>>f ;
    int brojac_neparnih(0), brojac_pozitivnih(0);
    if(a>0){brojac_pozitivnih++;if(a%2!=0)brojac_neparnih++;}
    if(b>0){brojac_pozitivnih++;if(b%2!=0)brojac_neparnih++;}
    if(c>0){brojac_pozitivnih++;if(c%2!=0)brojac_neparnih++;}
    if(d>0){brojac_pozitivnih++;if(d%2!=0)brojac_neparnih++;}
    if(e>0){brojac_pozitivnih++;if(e%2!=0)brojac_neparnih++;}
    if(f>0){brojac_pozitivnih++;if(f%2!=0)brojac_neparnih++;}
    if(brojac_pozitivnih==0)std::cout<<"Nema pozitivnog ni neparnog broja!";
    else{
        std::cout<<"Medju ovim brojevima ima "<<brojac_pozitivnih<<" pozitivnih
brojeva, a medju njima "<< brojac_neparnih<<" neparnih brojeva";
    }
    return 0;
}
```

9. Napišite C++ program koji za prirodan broj unesen sa tastature ispisuje sve njegove proste faktore, pri čemu svaki prosti faktor biće ispisan onoliko puta koliko puta učestvuje u tom broju. Na primjer, ako se unese broj 290472, program treba ispisati: 2 2 2 3 7 7 13 19, jer rastava ovog broja na proste faktore glasi $290472 = 2^3 \cdot 3 \cdot 7^2 \cdot 13 \cdot 19$.

Rješenje.

```
#include<iostream>
int main() {
    unsigned int n,i(2);
    std::cout<<"Unesite prirodan broj: "; std::cin>>n;
    if(n==1)std::cout<<"1";
```

```

else do{
    if(n%i==0){//ako je n djeljiv sa i
        int brojac(0);
        for(int j(1);j<=i;j++)if(i%j==0)brojac++;
        if(brojac==2)while(n%i==0){std::cout<<i<<" ";n/=i;}
    }
}
while(i++<=n);
return 0;
}

```

10. Napišite C++ program koji traži da se unese realan broj x i prirodan broj n , a zatim računa i ispisuje vrijednost sume

$$S = \sum_{k=1}^n \frac{(-1)^k}{x(x+k)}.$$

Na primjer, za $x = 2$ i $n = 5$, program treba ispisati -0.129762 . U programu nije dozvoljeno koristiti funkciju „pow“. Također, u programu ne smije biti više od jedne petlje.

Rješenje.

```

#include<iostream>
int main() {
    unsigned int n; double x, suma(0);
    std::cout<<"Unesite prirodan broj: "; std::cin>>n;
    std::cout<<"Unesite realan broj: "; std::cin>>x;
    for(int znak(-1), k(0); k<n; znak*=-1, k++) suma+=znak/(x*(x+k+1));
    std::cout<<suma;
    return 0;
}

```

11. Iz matematičke analize poznato je da za dovoljno veliko n vrijednost sume

$$S = \sum_{k=0}^n \frac{x^k}{k!}$$

približno je jednaka vrijednosti funkcije $F = e^x$, s obzirom da za $n \rightarrow \infty$ vrijednost sume S konvergira ka F (jer je sum S Taylorov razvoj funkcije F , koji konvergira za svaku vrijednost). Napišite program koji za zadanu vrijednost x određuje minimalnu vrijednost n takvu da se S i F poklapaju na barem prvih 5 decimala (tj. da je $|S - F| < 10^{-5}$). Program treba da ispiše nađenu vrijednost n , kao i odgovarajuće vrijednosti S i F (da bi se uvjerali da se one zaista poklapaju barem na prvih 5 decimala). Pri računanju sume S nije dozvoljeno posebno računati faktorijel niti koristiti funkciju „pow“ za računanje stepena, već treba iskoristiti ovisnost koja postoji između k -tog i $k + 1$ -vog člana sume.

Rješenje.

```
#include<iostream>
#include<cmath>
int main() {
    unsigned int n(1), broj(1); double x;
    std::cout<<"Unesite realan broj: "; std::cin>>x;
    double nx(x), suma(1), ex(exp(x));
    do{suma+=nx/broj;nx*=x; broj*=++n;}while(fabs(ex-suma)>=1e-5);
    std::cout<<"Broj n: "<<n<<" , Suma S: "<<suma<<" , Funkcija F: "<<ex;
    return 0;
}
```

12. Napišite C++ program koji traži unos realnih brojeva sa tastature sve dok korisnik ne unese broj 0. Nakon toga, program treba izračunati i ispisati aritmetičku i geometrijsku sredinu unesenih brojeva, ne uključujući unesenu nulu. Na primjer, ako korisnik unese brojeve 3, 8, 5.4, 2.13, 7 i 0, aritmetička sredina će biti 5.106, a geometrijska sredina 4.54168. Pri tome, program ne smije koristiti nizove.

Rješenje.

```
#include<iostream>
int main() {
    double x, suma(0), proizvod(1);
    int brojac(0);
    std::cout<<"Unesite cijele brojeve(0 za prekid): ";
    while(std::cin>>x,x!=0){brojac++;suma+=x;proizvod*=x;}
    std::cout<<"Aritmetička sredina je: "<<suma/brojac<< std::endl;
    std::cout<<"Geometrijska sredina je: "<<pow(proizvod,1.0/brojac);
    return 0;
}
```

13. Napišite C++ program koji traži unos dva cijela broja *a* i *b* sa tastature. Nakon toga, program treba ispisati sve savršene brojeve u opsegu od *a* do *b*. Broj je savršen ako je jednak sumi svih svojih djelilaca (ne uključujući njega samog). Na primjer, broj 6 je savršen jer su mu djelioци 1, 2 i 3, a $1 + 2 + 3 = 6$. Kao provjeru ispravnosti programa, možete koristiti činjenicu da je jedini savršeni broj u opsegu od 400 do 500 broj 496.

Rješenje.

```
#include<iostream>
int main() {
    int a,b;
    std::cout<<"Unesite vrijednost za opseg a i b(a<b):"; std::cin>>a>>b;
    for(int i(a);i<b;i++){
        int sav_broj(0);
        for(int j(1);j<i;j++)if(i%j==0)sav_broj+=j;
        if(sav_broj==i)std::cout<<sav_broj<<" ";
    }
    return 0;
}
```



```
}
```

14. Napišite C++ program koji traži unos prirodnog broja n sa tastature, a zatim ispisuje na ekranu da li je broj n palindrom ili nije. Broj je palindrom ako je jednak broju koji se dobije čitanjem njegovih cifara u obrnutom poretaku. Na primjer, broj 121 je palindrom. Za realizaciju programa nije dozvoljeno da se koristi niz.

Rješenje.

```
#include<iostream>
int main() {
    unsigned int n, palindrom(0);
    std::cout<<"Unesite cijeli broj: "; std::cin>>n;
    int kopija(n);
    do{palindrom=palindrom*10+kopija%10;}while((kopija/=10)!=0);
    if(n==palindrom)std::cout<<"Broj je polindrom!";
    else std::cout<<"Broj nije polindrom!";
    return 0;
}
```

15. Napišite C++ program koji traži da se sa tastature unesu cijeli brojevi n i m , a koji zatim iscrtava na ekranu pravougaonik sastavljen od zvjezdica čije su dužine stranica respektivno n i m . Na primjer, za $n = 15$ i $m = 5$, ispis na ekranu treba da izgleda kao

```
*****
*           *
*           *
*           *
*****
```

Rješenje.

```
#include<iostream>
int main() {
    int n,m;
    std::cout<<"Unesite n i m: "; std::cin>>n>> m;
    for(int i(0);i<m;i++){
        for(int j(0);j<n;j++){
            if(i==0 || j==0 || i==m-1 || j==n-1)std::cout<<"*";
            else std::cout<<" ";
            std::cout<<std::endl;
        }
    }
    return 0;
}
```

16. Napišite C++ program koji traži da se sa tastature unese cijeli broj n , a zatim iscrtava na ekranu jednakostranični trougao sastavljen od zvjezdica čija je osnovica horizontalna a vrh usmjeren nagore. Na primjer, ukoliko se unese $n = 6$, ispis na ekranu treba da izgleda kao

```
*
***
*****
*****
*****
*****
*****
```

Rješenje.

```
#include<iostream>
int main() {
    int n;
    std::cout<<"Unesite n :"; std::cin>>n;
    for(int i(0);i<n;i++){
        for(int j(0);j<2*n;j++){
            if(j>n-1-i && j<n+1+i)std::cout<<"*";
            else std::cout<<" ";
            std::cout<<std::endl;
        }
        return 0;
    }
}
```

17. Napišite C++ program koji traži da se sa tastature unese cijeli broj n , a zatim iscrtava na ekranu jednakostranični trougao čiji je rub sastavljen od zvjezdica čija je osnovica horizontalna a vrh usmjeren nagore, a čija je unutrašnjost prazna. Na primjer, ukoliko se unese $n = 6$, ispis na ekranu treba da izgleda kao

```
*
* *
*  *
*   *
*    *
*     *
*****
```

Rješenje.

```
#include<iostream>
int main() {
    int n;
    std::cout<<"Unesite n :"; std::cin>>n;
    for(int i(0);i<n;i++){
        for(int j(0);j<2*n;j++){
            if((j==n-i && i!=0 && i!=n-1)|| (j==n+i && i!=0 && i!=n-1)|| (i==0 && j==n)|| (i==n-1 && j>0))std::cout<<"*";
            else std::cout<<" ";
            std::cout<<std::endl;
        }
        return 0;
    }
}
```

18. Napišite C++ program koji traži od korisnika da unese prirodne brojeve m i n , pri čemu je $m > n$. Nakon toga, program treba na ekranu iscrtati figuru oblika " $<->$ " sastavljenu od zvjezdica, čija je širina m , a visina n znakova. Na primjer, za $m = 15$ i $n = 7$, prikaz na ekranu treba izgledati ovako:

```

      *          *
    *            *
  *              *
*****
  *              *
    *            *
      *          *

```

Rješenje.

```

#include<iostream>
int main() {
    int n, m;
    std::cout<<"Unesite m i n: "; std::cin>>m>>n ;
    if(n%2==0){std::cout<<"Broj n mora biti neparan!"; return 0;}
    for(int i(0);i<n;i++){
        for(int j(0);j<m;j++){
            if((i+j==(n-1)/2 && i!=(n-1)/2) || (i==(n-1)/2) || (j-i==m-(n+1)/2 && i!=(n-1)/2) || (i-j==(n-1)/2 && i!=(n-1)/2) || (i+j==m+(n-1)/2-1)) std::cout<<"*";
            else std::cout<<" ";
            std::cout<<std::endl;
        }
        return 0;
    }
}

```

19. Napišite C++ program koji traži od korisnika da unese prirodne brojeve m i n , pri čemu je $m > n$. Nakon toga, program treba na ekranu iscrtati figuru oblika " $>-<$ " sastavljenu od zvjezdica, čija je širina m , a visina n znakova. Na primjer, za $m = 15$ i $n = 7$, prikaz na ekranu treba izgledati ovako:

```

      *          *
    *            *
  *              *
*****
  *              *
    *            *
      *          *

```

Rješenje.

```

#include<iostream>
int main() {

```

```
int n, m;
std::cout<<"Unesite m i n: "; std::cin>>m>>n ;
if(n%2==0){std::cout<<"Broj n mora biti neparan!"; return 0;}
for(int i(0);i<n;i++){
    for(int j(0);j<m;j++){
        if((i==(n-1)/2 && (j>=(n-1)/2 && j<m-(n-1)/2))||(i==j && i<(n-1)/2)
        )||(i+j==m-1 && i<(n-1)/2)||(i+j==n-1 && i>(n-1)/2)||(j-i==m-n && i>(n-1)/2))std::cout<<"*";
        else std::cout<<" ";
        std::cout<<std::endl;
    }
    return 0;
}
```

20. Napišite C++ program koji traži od korisnika da unese cijele brojeve m i n . Nakon toga, program treba da iscrta figuru "pješčani sat" sastavljenu od zvjezdica, pri čemu su m i n redom širina i visina figure. Pretpostavite da je $n < m$ i $m < 20$. Također, radi jednostavnosti, pretpostavite da je visina n neparan broj. Na primjer, za $m = 10$ i $n = 7$ prikaz na ekranu treba izgledati ovako:

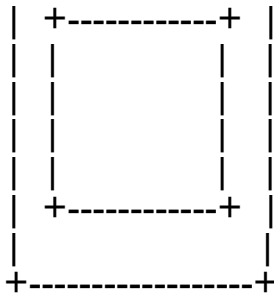
```
*****
*       *
*       *
*   *   *
*   *   *
*       *
*       *
*****
```

Rješenje.

```
#include<iostream>
int main() {
    int n, m;
    std::cout<<"Unesite m i n: "; std::cin>>m>>n ;
    if(n%2==0){std::cout<<"Broj n mora biti neparan!"; return 0;}
    for(int i(0);i<n;i++){
        for(int j(0);j<m;j++){
            if((i==0||i==n-1)||(i==j && i<=(n-1)/2 && i!=0)||(i+j==n-1 && i>(n-1)/2
            && i!=n-1)||(i+j==m-1 && i<=(n-1)/2 && i!=0)||(j-i==m-n && i>(n-1)/2 &&
            i!=n-1))std::cout<<"*";
            else std::cout<<" ";
            std::cout<<std::endl;
        }
        return 0;
    }
```

21. Napišite C++ program koji traži da se sa tastature unese prirodan broj n ne manji od 5 i ne veći od 24, a koji zatim na ekranu iscrtava šaru koja izgleda poput šare na sljedećoj slici, koja prikazuje primjer za $n = 10$:

```
+-----+
|           |
```



Preciznije, šara se sastoji od pravougaonika dimenzija $n \times 2$ znakova sastavljenog od znakova „+“, „-“, i „|“, unutar kojeg se nalazi drugi manji pravougaonik koji je od većeg pravougaonika razdvojen praznim slojem debljine jednog praznog mjesta po visini, a dva prazna mjesta po širini.

Rješenje.

```
#include<iostream>
int main() {
    int n;
    std::cout<<"Unesite n: "; std::cin>>n;
    for(int i(0);i<n;i++){
        for(int j(0);j<2*n;j++){
            if((i==0 && j==0)|| (i==0 && j==2*n-1)|| (i==n-1 && j==0)|| (i==n-1 &&
j==2*n-1)|| (i==2 && j==3)|| (i==2 && j==2*n-4)|| (i==n-3 && j==3)|| (i==n-3 &&
j==2*n-4)) std::cout<<"+";
            else if(i==0||i==n-1|| (i==2 && j>3 && j<2*n-4)|| (i==n-3 && j>3 && j<2*n-
4)) std::cout<<"-";
            else if(j==0||j==2*n-1|| (j==3 && i>2 && i<n-3)|| (j==2*n-4 && i>2 && i<n-
3)) std::cout<<"|";
            else std::cout<<" ";
            std::cout<<std::endl;
        }
        return 0;
    }
```

22. Napišite C++ program koji od korisnika prvo traži unos realnog broja x i stepena polinoma N . Zatim, program traži unos koeficijenata polinoma $a_N, a_{N-1}, \dots, a_1, a_0$, (počevši od koeficijenta uz najveći stepen). Nakon toga, program treba izračunati i ispisati vrijednost polinoma $P(x)$ za dati broj x , pri čemu je $P(x)$ ovako zadato $P(x) = \sum_{i=0}^N a_i x^i$. Program ne smije koristiti nizove, jer se vrijednost polinoma $P(x)$ može izračunati "u hodu", što znači da nije potrebno pamtit koeficijente. Najbolje rješenje neće koristiti funkciju za stepenovanje, s obzirom da se polinom može jednostavno izračunati bez stepenovanja pomoću **Hornerove sheme**.

Rješenje.

```
#include<iostream>
int main() {
    std::cout<<"Unesite realan broj x="; double x; std::cin>>x;
    std::cout<<"Unesite stepen polinoma: "; int N; std::cin>>N;
```

```
double suma(0);
for(int i(N); i>0; i--){
    std::cout<<"Unesite "<<i<<" koeficent polinoma: ";
    double koeficijent; std::cin>>koeficijent;
    suma+=koeficijent; suma*=x; // Hornerova sema
}
std::cout<<"Unesite slobodan clan: ";
double slobodan_clan; std::cin>>slobodan_clan;
std::cout<<"P(x)="<<suma+slobodan_clan;
return 0;
}
```

23. Napišite C++ program koji će prvo tražiti od korisnika da unese prirodan broj N , a zatim N realnih brojeva a_1, a_2, \dots, a_N . Nakon toga, program treba da izračuna i ispiše vrijednost izraza

$$\frac{1}{a_1} + \frac{1}{a_1 + a_2} + \frac{1}{a_1 + a_2 + a_3} + \dots + \frac{1}{a_1 + a_2 + a_3 + \dots + a_N}.$$

Program ne smije koristiti niz.

Rješenje.

```
#include<iostream>
int main() {
    std::cout<<"Unesite N: "; unsigned int N; std::cin>>N;
    double suma_elementa{0}, suma{0};
    for(int i(0); i<N; i++){
        std::cout<<"Unesite "<<i+1<<" broj: "; double broj; std::cin>>broj;
        suma_elementa+=broj; suma+=1/suma_elementa;
    }
    std::cout<<"Trazena suma je : "<<suma;
    return 0;
}
```

24. Napišite C++ program koji traži unos prirodnog broja n . Nakon toga, program treba izračunati i ispisati vrijednost verižnog razlomka definiranog kao

$$Y = \frac{1}{1 + \frac{1}{2 + \frac{1}{3 + \frac{1}{\dots + \frac{1}{n}}}}}$$

Na primjer, za $n = 3$, program treba ispisati 0.7, jer je

$$Y = \frac{1}{1 + \frac{1}{2 + \frac{1}{3}}} = \frac{7}{10} = 0.7$$

Rješenje.

```
#include<iostream>
int main() {
    std::cout<<"Unesite prirodan broj: "; unsigned int n; std::cin>>n;
    double suma(0);
    for(int i(n); i>0;i--) suma=1/(suma+i);
    std::cout<<"Trazena suma je: "<<suma;
    return 0;
}
```

25. Napišite C++ program koji traži unos dva prirodna broja n i m sa tastature. Nakon toga, program treba ispisati sve proste brojeve između n i m , uključujući i same granice n i m .

Rješenje.

```
#include<iostream>
int main() {
    std::cout<<"Unesite prirodne brojeve n i m: ";
    unsigned int n, m; std::cin>>n>>m;
    if(n>m){int pom(m);m=n;n=pom;}
    for(int brojac(0), i(n); i<=m; i++, brojac=0){
        for(int j(1);j<=i;j++)if(i%j==0)brojac++;
        if(brojac==2)std::cout<<i<<" ";
    }
    return 0;
}
```

26. Napišite C++ program koji traži od korisnika unos prirodnog broja n sa tastature, a zatim pronalazi i ispisuje n -ti prosti broj. Na primjer, ako korisnik unese $n = 30$, program treba ispisati 113, jer je 113 trideseti po redu prosti broj.

Rješenje.

```
#include<iostream>
int main() {
    unsigned int n, brojac(0);
    std::cout<<"Unesite broj n: "; std::cin>>n;
    for(int brojac1(0), i(2); i>0; i++, brojac1=0){
        for(int j(1);j<=i;j++)if(i%j==0)brojac1++;
        if(brojac1==2 && ++brojac==n){
            std::cout<<n<<"-ti po redu prost broj je: "<<i; break;
        }
    }
    return 0;
}
```

27. Napišite C++ program koji traži unos dva prirodna broja a i b sa tastature. Nakon toga, program treba ispisati broj brojeva u rasponu od a do b (uključujući i njih) koji su djeljivi sa sumom svojih cifara. Na primjer, broj 10 je takav broj, jer

je djeljiv sa $1 + 0 = 1$. Testirajte program koristeći neke karakteristične vrijednosti koje su prikazane u donjoj tabeli.

<i>a</i>	<i>b</i>	<i>rezultat</i>
10	50	14
100	1000	181
351	351	1
352	353	0
10000	20000	1417
100000	200000	11167

Rješenje.

```
#include<iostream>
int main() {
    std::cout<<"Unesite brojeve a i b: "; int brojac(0), a, b; std::cin>>a>>b;
    if(a>b){int pom(b); b=a; a=pom;}
    for(int pom(a), suma_cifara(0), i(a); i<=b; i++, suma_cifara=0, pom=i){
        do{suma_cifara+=pom%10;}while((pom/=10)!=0);
        if(i%suma_cifara==0)brojac++;
    }
    std::cout<<brojac;
    return 0;
}
```

28. Napišite C++ program koji traži unos dva prirodna broja m_1 i m_2 sa tastature. Nakon toga, program treba ispisati tablicu sa dvije kolone. Prva kolona treba sadržavati prirodne brojeve od m_1 do m_2 uključujući ih, dok druga kolona treba sadržavati sumu svih djelilaca odgovarajućih brojeva iz prve kolone. Na primjer, ako je u prvoj koloni broj 12, u drugoj koloni treba biti broj 28, jer djelioци broja 10 su: 1, 2, 3, 4, 6 i 12, a njihova suma je $1+2+3+4+6+12=28$. Formatiranje tablice možete prilagoditi prema vlastitom izboru, ali važno je da ispisani brojevi budu lijepo poravnati uz desnu ivicu.

Rješenje.

```
#include<iostream>
#include<iomanip>
int main() {
    std::cout<<"Unesite brojeve m1 i m2: ";
    unsigned int m1, m2; std::cin>>m1>>m2;
    if(m1>m2){int pom(m2); m2=m1; m1=pom;}
    for(int i(m1), suma(0); i<=m2; i++, suma=0){
        for(int j(1); j<=i; j++) if(i%j==0) suma+=j;
        std::cout<<i<<std::setw(5)<<std::right<<suma<<std::endl;
    }
    return 0;
}
```


29. U C++-u napišite funkciju f koja ima dva parametra a i n , te računa a^n bez korištenja funkcije „**std::pow**“ iz biblioteke „**cmath**“. Ova funkcija treba podržavati cijele brojeve proizvoljnog znaka, uključujući i negativne brojeve. Na primjer, za sljedeće pozive funkcije f unutar glavne funkcije „**main**“

```
std::cout<<f(2,6)<<" "<<f(10,-3)<<" "<<f(5);;
```

na ekranu trebaju biti ispisani brojevi $64(2^6)$, $0.001(10^{-3})$ i $25(5^2)$.

Rješenje.

```
#include<iostream>
double f(int a, int n=2){
    double rezultat(1); bool negativan(false);
    if(n<0){n*=-1; negativan=true;}
    for(int i(1); i<=n; i++)rezultat*=a;
    return (!negativan? rezultat : 1.0/rezultat);
}
int main(){
    std::cout<<f(2,6)<<" "<<f(10,-3)<<" "<<f(5);
    return 0;
}
```

30. U C++-u napišite funkciju f koja koristi Heronov postupak za računanje kvadratnog korijena proizvoljnog broja x . Heronov postupak generira niz brojeva a_0, a_1, a_2, \dots po sljedećem pravilu

$$a_0 = 1; a_{k+1} = \frac{a_k + \frac{x}{a_k}}{2}, k > 0.$$

Budući da niz a_k brzo konvergira brzo ka korijenu iz x , dovoljno je u praksi izračunati samo nekoliko elemenata ovog niza, jer se vrlo brzo elementi počinju praktično ponavljati (sa onolikom tačnošću koliku dopušta realni tip podataka). Funkcija f treba koristiti ovaj postupak i vratiti posljednju izračunatu vrijednost a_k . Testirajte napisanu funkciju f u glavnom programu koji traži unos broja x sa tastature i ispisuje njegov korijen preko poziva $f(x)$, kao i korijen korištenjem ugrađene funkcije „**std::sqrt**“. Naravno, rezultati trebaju biti isti. Funkcija f treba biti implementirana bez korištenja nizova.

Rješenje.

```
#include<iostream>
#include<cmath>
double f(double x){
    double stari_broj(1);
    while(true){
        double novi_broj{(stari_broj+x/stari_broj)/2};
        if(novi_broj==stari_broj) return novi_broj;
        stari_broj=novi_broj;
    }
}
```

```

}
int main() {
    std::cout<<"sqrt: "<<std::sqrt(20)<<" korijen: "<<f(20);
    return 0;
}

```

31. Napišite C++ program koji za prirodan broj n unesen sa tastature ispisuje na ekran formatiranu tablicu množenja za sve brojeve od 1 do n uključivo. Na primjer, ukoliko se unese $n = 4$, ispis na ekranu bi trebao izgledati poput sljedećeg:

```

+----+----+----+----+
|  1|  2|  3|  4 |
+----+----+----+----+
|  2|  4|  6|  8 |
+----+----+----+----+
|  3|  6|  9| 12 |
+----+----+----+----+
|  4|  8| 12| 16 |
+----+----+----+----+

```

Pretpostavite da je n takav da će čitava tablica stati na ekranu i da niti jedan produkt neće imati više od 3 cifre. Za ispis koristite objekat izlaznog toka **cout**, a za prilagođavanje širine ispisa manipulator **setw**.

Rješenje.

```

#include<iostream>
#include<iomanip>
int main() {
    std::cout<<"Unesite broj n: "; unsigned int n; std::cin>>n;
    for(int k{1};k<=n;k++){std::cout<<"<< "----";} std::cout<<"\n";
    for(int i{1};i<=n;i++){
        for(int j{1};j<=n;j++)std::cout<<"| "<<std::setw(5)<<i*j;std::cout<<"|\n";
        for(int k{1};k<=n;k++)std::cout<<"<< "----";std::cout<<"\n";
    }
    return 0;
}

```

32. U C++-u napišite funkciju f bez parametara, koja od korisnika traži unos nekog realnog broja x . Ako je zaista unesen realan broj, funkcija f treba vratiti taj broj kao rezultat. Ako nije unesen broj, funkcija treba ispisati "Neispravan unos, pokušajte ponovo." i ponovno tražiti unos broja, sve dok se ne unese ispravan broj. Potom upotrijebite funkciju f u glavnom programu (**main** funkcija) koji traži od korisnika unos koeficijenata a , b i c linearne jednačine $ax + b = c$. Program zatim pronalazi i ispisuje rješenje te jednačine, pod uslovom da postoji ili odgovarajuću poruku u suprotnom.

Rješenje.

```
#include<iostream>
double f() {
    while(true) {
        double x;
        if(!(std::cin>>x)){std::cout<<"Neispravan unos, pokušajte ponovo.\n";
            std::cin.clear();std::cin.ignore(1000, '\n');//deblokira se i cisti tok}
        else return x;
    }
}
int main() {
    std::cout<<"Unesite koeficijente a, b i c jednacine ax+b=c: ";
    double a(f()),b(f()),c(f());
    if(a==0&&(b!=c))std::cout<<"Nema rjesenja";
    else if(a==0&&(b==c))std::cout<<"Jednacina ima besk. mnogo rjesenja.";
    else std::cout<<"Rjesenje jednacine je: "<<(c-b)/a;
    return 0;
}
```

33. U C++-u napišite funkciju f koja prima jedan cjelobrojni parametar n , te vraća logičku vrijednost „true“ ako je broj n simetričan (isto se čita sa obje strane, npr. broj 121 je simetričan), inače vraća vrijednost „false“. Nakon toga, demonstrirajte funkciju f u glavnom programu gdje korisnik unosi brojeve sa tastature. Program treba za svaki uneseni broj ispisati da li je simetričan ili ne. Program završava rad kada se unese broj nula.

Rješenje.

```
#include<iostream>
bool f(int n){
    int simetrican(0),kopija(n);
    do{simetrican=simetrican*10+kopija%10;}while((kopija/=10)!=0);
    return n==simetrican;
}
int main() {
    while(1){
        std::cout<<"\nUnesite prirodan broj (0-kraj): ";
        if(int n;std::cin>>n,n!=0){
            if(f(n))std::cout<<"Broj je simetrican.";
            else std::cout<<"Broj nije simetrican.";
        }
        else break;
    }
    return 0;
}
```

34. Napišite funkciju f koja kao parametar prima neki prirodan broj i koja vraća logičku vrijednost „true“ ili „false“ u zavisnosti da li u broju koji je prosljeđen kao parametar ima jednakih cifara ili ne. Nije dopušteno korištenje nizova. Napišite kratki testni program (**main** funkcija) u kojem ćete demonstrirati kako se poziva napisana funkcija i kako se može iskoristiti njen rezultat.

Rješenje.

```
#include<iostream>
bool f(unsigned int n){
    do{
        for(int kopija(n/10); kopija!=0;){
            if((n%10)==(kopija%10))return true;
            kopija/=10;
        }
    }while(n/=10);
    return false;
}
int main(){
    std::cout<<"Unesite prirodan broj : ";int broj;std::cin>>broj;
    if(f(broj))std::cout<<"Broj ima jednakih cifara.";
    else std::cout<<"Broj nema jednakih cifara.";
    return 0;
}
```

35. Napišite funkciju f koja kao parametar prima neki prirodan broj i koja vraća logičku vrijednost „true“ ili „false“ u zavisnosti da li su u broju koji je prosljeđen kao parametar sve cifre različite ili ne. Nije dopušteno korištenje nizova. Napišite i kratki testni program u kojem ćete demonstrirati kako se poziva napisana funkcija i kako se može iskoristiti njen rezultat.

Rješenje.

```
#include<iostream>
bool f(unsigned int n){
    do{
        for(int kopija(n/10); kopija!=0;){
            if((n%10)==(kopija%10))return true;
            kopija/=10;
        }
    }while(n/=10);
    return false;
}
int main(){
    std::cout<<"Unesite prirodan broj : ";int broj;std::cin>>broj;
    if(f(broj))std::cout<<"Sve cifre nisu medjusobno razlicite.";
    else std::cout<<"Sve cifre su medjusobno razlicite.";
    return 0;
}
```

36. Napišite funkciju f koja ima kao parametar neki prirodan broj, a koja daje kao rezultat isti taj broj, samo iz kojeg su izbačene sve cifre koje su nule. Na primjer, ukoliko se kao parametar funkciji prosljedi broj 12030040, rezultat funkcije treba da bude 1234. Ukoliko broj ne sadrži niti jednu nulu kao svoju cifru, onda je on sam ujedno i rezultat funkcije. Napisanu funkciju demonstrirati u kratkom testnom programu na brojevima koji se unose sa tastature. Program treba da završi rad kada se kao broj unese 0.

Rješenje.

```
#include<iostream>
int f(unsigned int n){
    int broj(0),baza(1);
    do{
        if(int ostatak(n%10);ostatak!=0){broj+=baza*ostatak;baza*=10;}
    }while(n/=10);
    return broj;
}
int main(){
    std::cout<<"\nUnesite prirodan broj(0-kraj): "; int n;
    while(std::cin>>n,n!=0){
        std::cout<<"Broj bez nula je: "<<f(n)<<"\n";
        std::cout<<"\nUnesite prirodan broj(0-kraj): ";
    }
    return 0;
}
```

37. Napišite funkciju g sa dva parametra n i $parnost$, od kojih je prvi cjelobrojnog a drugi logičkog tipa. Ukoliko parametar $parnost$ ima vrijednost „true“, funkcija treba vratiti broj koji se sastoji samo od parnih cifara broja n (uzetih u istom poretku), a ukoliko parametar $parnost$ ima vrijednost „false“, funkcija treba vratiti broj koji se sastoji samo od neparnih cifara broja n . Na primjer, nakon izvršenja naredbi

```
std::cout<<g(3427816,true)<<std::endl;
std::cout<<g(3427816,false)<<std::endl;
```

treba da se ispišu brojevi 4286 i 371. Napisanu funkciju trebate demonstrirati na kratkom testnom programu koji će za broj unesen sa tastature ispisati brojeve sastavljene od njegovih parnih odnosno neparnih cifara. Nije dopušteno korištenje nizova. Proces treba da se ponavlja sve dok korisnik ne unese nulu.

Rješenje.

```
#include<iostream>
int g(int n, bool parnost){
    int broj(0),baza(1);
    do{
        int ostatak(n%10);
        if(parnost==true && (ostatak%2==0)){broj+=baza*ostatak;baza*=10;}
        else if(parnost==false && (ostatak%2==1)){broj+=baza*ostatak;baza*=10;}
    }while(n/=10);
    return broj;
}
int main(){

    return 0;
}
```

38. Neka je n prirodan broj, i neka je n_1 proizvod cifara broja n , n_2 proizvod cifara broja n_1 , n_3 proizvod cifara broja n_2 , itd. Najmanji broj k za koji vrijedi da je n

jednocifren broj u teoriji brojeva se naziva **multiplikativna otpornost broja**. Na primjer, multiplikativna otpornost broja 6788 je 6, jer važi:

$$n_1 = 6 \cdot 7 \cdot 8 \cdot 8 = 2688$$

$$n_2 = 2 \cdot 6 \cdot 8 \cdot 8 = 768$$

$$n_3 = 7 \cdot 6 \cdot 8 = 336$$

$$n_4 = 3 \cdot 3 \cdot 6 = 54$$

$$n_5 = 5 \cdot 4 = 20$$

$$n_6 = 2 \cdot 0 = 0$$

Napišite funkciju **mult_otpornost** koja kao parametar prima prirodan broj n , a kao rezultat vraća njegovu multiplikativnu otpornost. Napisanu funkciju demonstrirati na kratkom testnom programu koji će ispisati sve brojeve u opsegu od a do b uključivo koji imaju zadanu multiplikativnu otpornost k , pri čemu se vrijednosti a , b i k zadaju preko tastature. Na primjer, ukoliko se za a , b i k unesu vrijednosti 9985, 10002 i 3, treba da se ispišu brojevi 9987, 9988, 9989, 9992, 9993, 9994, 9995, 9998 i 9999, jer ti brojevi u zadanom intervalu imaju multiplikativnu otpornost 3. Također, za sve razumne vrijednosti n , multiplikativna otpornost ima uglavnom jednocifrenu vrijednost, pa o tome vodite računa kada budete testirali program. Brojevi koji imaju multiplikativnu otpornost veću od 9 veoma su rijetki.

Rješenje.

```
#include<iostream>
int mult_otpornost(unsigned int n){
    for(int i(1),produkt(1);;produkt=1,i++){
        do{produkt*=n%10;}while(n/=10);
        if(produkt<=9) return i;
        n=produkt;
    }
}
int main(){
    std::cout<<"Unesite brojeve a i b: "; unsigned int a,b; std::cin>>a>>b;
    std::cout<<"Unesite multipl. otpornost k: "; unsigned int k; std::cin>>k;
    std::cout<<"Brojevi od "<<a<<" do "<<b<<" sa multip. otpor. "<<k<<" su: ";
    for(unsigned int i{a};i<b;i++)if(mult_otpornost(i)==k)std::cout<<i<<" ";
    return 0;
}
```

39. Napišite program koji traži da se sa tastature unese jedan kompleksan broj z i dva realna broja x i y , a koji zatim računa i ispisuje vrijednost izraza

$$\left(x + \frac{5i}{z^3 \sqrt{x - yi}}\right)^2.$$

Na primjer, ukoliko se sa tastature unesu podaci $z = (3, 7)$, $x = 2$ i $y = 6$, program treba da ispiše rezultat (4.004757, 0.0173699) jer je

$$\left(2 + \frac{5i}{(3+7i)^3\sqrt{2-6i}}\right)^2 = 4.004757 - 0.0173699i.$$

Rješenje.

```
#include<iostream>
#include<complex>
#include<cmath>
int main() {
    std::cout<<"Unesite realni i imaginarni dio kompleksnog broja z: ";
    double re,im; std::cin>>re>>im; std::complex<double>z(re,im);
    std::cout<<"Unesite dva realna broja x i y: "; double x,y; std::cin>>x>>y;
    std::cout<<
    (std::complex<double>(x,0)+(std::complex<double>(0,5)/(std::pow(z,3)*std::s
    qrt(std::complex<double>(x,-
    y))))*(std::complex<double>(x,0)+(std::complex<double>(0,5)/(std::pow(z,3)
    *std::sqrt(std::complex<double>(x,-y)))));
    return 0;
}
```

40. Poznata **Moievrova teorema** za kompleksne brojeve tvrdi da za sve realne brojeve x i n važi

$$(\cos x + i \sin x)^n = \cos nx + i \sin nx.$$

Napišite program koji će za realne brojeve x i n unesene sa tastature izračunati lijevu i desnu stranu ovog izraza i ispisati ih na ekranu, koristeći kompleksni tip podataka. Na nekoliko primjera provjerite da li se zaista dobijaju jednake vrijednosti.

Rješenje.

```
#include<iostream>
#include<complex>
#include<cmath>
int main() {
    std::cout<<"Unesite realne brojeve x i n: "; double x, n; std::cin>>x>>n;
    std::cout<<std::pow(std::complex<double>(cos(x),sin(x)),n)<<std::endl;
    std::cout<<std::complex<double>(cos(n*x),sin(n*x))<<std::endl;
    return 0;
}
```

41. Napišite program za rješavanje kvadratne jednačine tako da koristi kompleksne promjenljive, pri čemu koeficijenti kvadratne jednačine a , b i c trebaju biti kompleksni brojevi. Rješenja čiji je imaginarni dio jednak nuli treba ispisati tako da se ispisuju kao realni brojevi, a ne kao uređeni parovi realnih brojeva.

Rješenje.

```
#include<iostream>
#include<complex>
typedef std::complex<double>Komp;
int main() {
    std::cout<<"Unesi koeficijente a, b, i c:\n"; double r1,im1,r2,im2,r3,im3;
```

```
std::cin>>r1>>im1>>r2>>im2>>r3>>im3; Komp a(r1,im1),b(r2,im2),c(r3,im3);
auto d(b*b-std::complex<double>(4,0)*a*c);
auto x1((-b-std::sqrt(d))/(Komp(2,0)*a));
auto x2((-b+std::sqrt(d))/(Komp(2,0)*a));
if(x1.imag()==0&&x2.imag()==0)
    std::cout<<"x1="<<x1.real()<<"\nx2="<<x2.real();
else std::cout<<"x1="<<x1<<"\nx2="<<x2;
return 0;
}
```

42. Napišite funkcije f i g koje kao parametre redom primaju vektor realnih i vektor cijelih brojeva, a vraćaju kao rezultat logičku vrijednost. Funkcija f vraća vrijednost „true“ ako su svi elementi vektora međusobno identični, inače vraća „false“. Funkcija g vraća vrijednost „true“ ako su svi elementi vektora međusobno različiti, inače vraća „false“. Nije dozvoljeno da se radi sortiranje elemenata vektora. Bilo bi poželjno da funkcije f i g su linearne vremenske složenosti. Napisane funkcije testirajte u glavnom programu (**main** funkciji) u kojem se prvo sa tastature unosi prirodan broj n i n elemenata vektora, a nakon toga se pozivaju napisane funkcije sa ciljem da se utvrdi da li su svi elementi vektora jednaki odnosno različiti.

Rješenje.

```
#include<iostream>
#include<vector>
bool f(const std::vector<double>&v){
    for(int i(0);i<v.size()-1;i++)if(v[i]!=v[i+1])return false;
    return true;
}
bool g(const std::vector<int>&v){
    int maska{0};
    for(int e:v){
        int bit{1<<e};//oper. << radi pomjeranje u lijevo za e mjesta
        if(maska&bit)return false;//broj se ponavlja
        maska|=bit;//oper. |= postavlja bit na masku
    }
    return true;
}
int main(){
    std::vector<double>v1{2, 2, 2, 2, 2};
    std::vector<int>v2{1,5,2,1,3,1,4,2,3,5};
    if(f(v1))std::cout<<"Svi elementi u v1 su identicni.\n";
    else std::cout<<"\nNeki elementi u v1 se ponavljaju.";
    if(g(v2))std::cout<<"\nSvi elementi u v2 su razliciti.";
    else std::cout<<"Nisu svi elementi u v2 razliciti.";
    return 0;
}
```

43. Napišite funkciju f koja kao parametar prima vektor cijelih brojeva, te vraća logičku vrijednost „true“ ili „false“ u zavisnosti da li u vektoru ima parnih brojeva ili ne. Nije dopušteno koristiti operator `%` niti operatore `(==)` i `(!=)` za poređenje

na jednakost odnosno različitost. Napisanu funkciju demonstrirajte u testnom programu u kojem se sa tastature unosi prirodan broj n , a zatim n cijelih brojeva, koji se unose u vektor. Nakon unosa, program treba pozvati napisanu funkciju sa ciljem da utvrdi da li među unesenim brojevima ima parnih brojeva ili ne i ispisati odgovarajuću poruku, ovisno od rezultata funkcije.

Rješenje.

```
#include<iostream>
#include<vector>
bool f(const std::vector<int>&v) {
    for(int i{0};i<v.size();i++)if(!(v[i]&1))return true;//& je bitovsko AND
    return false;
}
int main() {
    std::cout<<"Unesite velicinu vektora (n): "; int n; std::cin>>n;
    std::cout<<"Unesite elemente vektora: "; std::vector<int>v(n);
    for(int i{0},broj;i<n;i++){std::cin>>broj; v[i]=broj;}
    if(f(v))std::cout<<"U vektoru ima parnih brojeva.";
    else std::cout<<"U vektoru nema parnih brojeva.";
    return 0;
}
```

44. Napišite funkciju f koja kao parametar prima vektor cijelih brojeva, te vraća logičku vrijednost „true“ ili „false“ u zavisnosti da li u vektoru ima potpunih kvadrata (tj. brojeva koji se mogu napisati kao kvadrati nekog drugog prirodnog broja) ili ne. Napisanu funkciju demonstrirajte u testnom programu u kojem se sa tastature unosi prirodan broj n , a zatim n cijelih brojeva, koji se unose u vektor. Nakon unosa, program treba pozvati napisanu funkciju sa ciljem da utvrdi ima li među unesenim brojevima potpunih kvadrata ili ne i ispisati odgovarajuću poruku, ovisno od rezultata funkcije.

Rješenje.

```
#include<iostream>
#include<vector>
#include<cmath>
bool f(std::vector<int>&v) {
    for(int i{0};i<v.size();i++){
        if(v[i]<0)v[i]*=-1; int korijen(std::sqrt(v[i]));
        if(korijen*korijen==v[i])return true;
    }
    return false;
}
int main() {
    std::cout<<"Unesite velicinu vektora (n): "; int n; std::cin>>n;
    std::cout<<"Unesite elemente vektora: "; std::vector<int>v(n);
    for(int i{0},broj;i<n;i++){std::cin>>broj; v[i]=broj;}
    if(f(v))std::cout<<"U vektoru ima potpunih kvadrata.";
    else std::cout<<"U vektoru nema potpunih kvadrata.";
    return 0;
}
```

45. Napišite funkciju f koja ima dva parametra, od kojih je prvi niz realnih brojeva, a drugi broj elemenata u nizu. Funkcija treba da kreira vektor realnih brojeva koji sadrži isti broj elemenata kao i zadani niz, da prepíše sve elemente niza u kreirani vektor u obrnutom poretku, i da vrati tako popunjeni vektor kao rezultat. Napisanu funkciju demonstrirajte u glavnom programu koji prvo sa tastature unosi elemente niza kapaciteta 10 elemenata. Nakon toga, program poziva napisanu funkciju sa ciljem generiranja vektora čiji su elementi u obrnutom poretku u odnosu na elemente unesenog niza. Na kraju, program ispisuje elemente generiranog vektora na ekran (međusobno razdvojene zarezom).

Rješenje.

```
#include<iostream>
#include<vector>
std::vector<double>f(double A[], int n){
    std::vector<double>v(n);
    for(int i{n-1};i>=0;i--)v[n-1-i]=A[i];
    return v;
}
int main(){
    double A[10];
    std::cout<<"Unesite elemente niza A: ";
    for(int i{0};i<10;i++)std::cin>>A[i];
    auto v(f(A,10)); std::cout<<"Elementi vektora u obrnutom poretku:\n";
    for(int i{0};i<v.size()-1;i++)std::cout<<v[i]<<" ";
    std::cout<<v[v.size()-1];
    return 0;
}
```

46. Napišite funkciju f koja kao parametar prima vektor cijelih brojeva, te vraća logičku vrijednost „true“ ili „false“, u zavisnosti da li u vektoru ima trocifrenih brojeva ili ne. Napisanu funkciju demonstrirajte u testnom programu u kojem se sa tastature unosi prirodan broj n , a zatim n cijelih brojeva, koji se unose u vektor. Nakon unosa, program treba pozvati napisanu funkciju sa ciljem da utvrdi ima li među unesenim brojevima trocifrenih brojeva ili ne i ispisati odgovarajuću poruku, ovisno od rezultata funkcije.

Rješenje.

```
#include<iostream>
#include<vector>
bool f(const std::vector<int>&v){
    for(int e:v){//rasponska for petlja
        int brojac{1},k(e);
        while(k/=10)brojac++;
        if(brojac==3) return true;
    }
    return false;
}
int main(){
```

```
std::cout<<"Unesite velicinu vektora (n): "; int n; std::cin>>n;
std::cout<<"Unesite elemente vektora: "; std::vector<int>v(n);
for(int i{0}, broj; i<n; i++){std::cin>>broj; v[i]=broj;}
if(f(v))std::cout<<"U vektoru ima trocifrenih brojeva.";
else std::cout<<"U vektoru nema trocifrenih brojeva.";
return 0;
}
```

47. Napišite funkciju f koja kao parametar prima vektor cijelih brojeva, te vraća logičku vrijednost „true“ ili „false“, u zavisnosti da li u vektoru ima brojeva koji su stepeni broja 2 ili ne. Nije dopušteno korištenje logaritamske funkcije `std::log`. Napisanu funkciju demonstrirajte u testnom programu u kojem se sa tastature unosi prirodan broj n . a zatim n cijelih brojeva, koji se unose u vektor. Nakon unosa, program treba pozvati napisanu funkciju sa ciljem da utvrdi da li među unesenim brojevima ima onih stepena dvojke ili ne i ispisati odgovarajuću poruku, ovisno od rezultata funkcije.

Rješenje.

```
#include<iostream>
#include<vector>
bool f(const std::vector<int>&v) {
    for(int e:v){//raspona for petlja
        int k(e); while(!(k%2))k/=2; if(k==1)return true;
    }
    return false;
}
int main() {
    std::cout<<"Unesite velicinu vektora (n): "; int n; std::cin>>n;
    std::cout<<"Unesite elemente vektora: "; std::vector<int>v(n);
    for(int i{0}, broj; i<n; i++){std::cin>>broj; v[i]=broj;}
    if(f(v))std::cout<<"U vektoru ima brojeva stepena dvojke.";
    else std::cout<<"U vektoru nema brojeva stepena dvojke.";
    return 0;
}
```

48. Napišite funkciju f koja kao parametar prima jedan prirodan broj, a kao rezultat vraća vektor čiji su elementi svi djeloci broja zadanog kao parametar. Napisanu funkciju demonstrirajte u glavnom programu u kojem se sa tastature unosi prirodan broj n i koji nakon toga poziva napisanu funkciju sa ciljem generiranja vektora čiji su elementi djeloci broja n , čije elemente na kraju treba ispisati na ekranu (međusobno razdvojene razmakom).

Rješenje.

```
#include<iostream>
#include<vector>
std::vector<unsigned int>f(unsigned int n) {
    std::vector<unsigned int>v;
    for(int i{1}; i<=n; i++){if(!(n%i))v.push_back(i);}
    return v;
}
```

```
}  
int main() {  
    std::cout<<"Unesite broj (n): "; unsigned int n; std::cin>>n;  
    auto v(f(n)); std::cout<<"Elementi vektora su: ";  
    for(unsigned int e: v) std::cout<<e<<" ";  
    return 0;  
}
```

49. Napišite funkciju f koja kao parametar prima jedan prirodan broj, a kao rezultat vraća vektor čiji su elementi svi stepeni broja 2, a koji su manji od broja zadanog kao parametar. Nije dopušteno koristiti funkciju `std::pow`. Napisanu funkciju demonstrirajte u glavnom programu u kojem se sa tastature unosi prirodan broj i koji nakon toga poziva napisanu funkciju sa ciljem generiranja vektora čiji su elementi svi stepeni dvojke manji od n , čije elemente na kraju treba ispisati na ekran (međusobno razdvojene razmakom).

Rješenje.

```
#include<iostream>  
#include<vector>  
std::vector<unsigned int>f(unsigned int n) {  
    std::vector<unsigned int>v;  
    for(int produkt{1};produkt<n;produkt*=2)v.push_back(produkt);  
    return v;  
}  
int main() {  
    std::cout<<"Unesite broj (n): "; unsigned int n; std::cin>>n;  
    auto v(f(n)); std::cout<<"Elementi vektora stepena dvojke su: ";  
    for(unsigned int e: v) std::cout<<e<<" ";  
    return 0;  
}
```

50. Napišite funkciju f koja kao parametar prima jedan prirodan broj, a kao rezultat vraća vektor čiji su elementi potpuni kvadrati, a koji su manji od broja zadanog kao parametar. Napisanu funkciju demonstrirajte u glavnom programu u kojem se sa tastature unosi prirodan broj i koji nakon toga poziva napisanu funkciju sa ciljem generiranja vektora čiji su elementi svi potpuni kvadrati manji od n , čije elemente na kraju treba ispisati na ekran (međusobno razdvojene razmakom).

Rješenje.

```
#include<iostream>  
#include<vector>  
#include<cmath>  
std::vector<unsigned int>f(unsigned int n) {  
    std::vector<unsigned int>v;  
    for(int i{1};i<n;i++)  
        if(int(std::sqrt(i))*int(std::sqrt(i))==i)v.push_back(i);  
    return v;  
}  
int main() {  
    std::cout<<"Unesite broj (n): "; unsigned int n; std::cin>>n;
```

```
auto v(f(n)); std::cout<<"Potpuni kvadrati su: ";  
for(unsigned int e: v)std::cout<<e<<" ";  
return 0;  
}
```

ELEKTRONSKO IZDANJE