

PROGRAMIRANJE

ZBIRKA ZADATAKA

Ishod učenja 3

2023-2024.

Zadatak 1

Napišite funkciju koja prima iznos temperature i vraća pozivatelju tekst prognoze prema pravilima:

- Ako je primljena vrijednost temperature između 15 i 30 stupnjeva, vratite tekst prognoze „Taman“.
- Ako je primljena vrijednost temperature iznad 30 stupnjeva, vratite tekst prognoze „Prevrucе“.
- Ako je primljena vrijednost temperature ispod 15 stupnjeva, vratite tekst prognoze „Prehladno“.

Pozovite funkciju iz main()-a tri puta kako biste demonstrirali sva tri slučaja i ispišite vraćene tekstove.

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

string pretvori_u_tekst(double temperatura) {
    if (temperatura > 30.0) {
        return "Prevrucе";
    }
    else if (temperatura < 15.0) {
        return "Prehladno";
    }
    else {
        return "Taman";
    }
}

int main() {
    cout << pretvori_u_tekst(7.5) << endl;
    cout << pretvori_u_tekst(32.0) << endl;
    cout << pretvori_u_tekst(22.5) << endl;

    return 0;
}
```

Zadatak 2

Napišite funkciju koja prima prirodni broj N i vraća je li broj prost ili nije. Pozovite funkciju za sve brojeve između 1 i 500 te ispišite samo one proste.

Moguće rješenje:

```
#include <iostream>
using namespace std;

bool je_li_prost(int broj) {
    for (int i = 2; i < broj; i++) {
        if (broj % i == 0) {
            return false;
        }
    }
    return true;
}

int main() {
    for (int i = 1; i <= 500; i++) {
        if (je_li_prost(i)) {
            cout << i << " ";
        }
    }
    cout << endl;
    return 0;
}
```

Zadatak 3

Napišite funkciju koja prima formulu molekule i ispisuje količinu atoma kisika (element oznake O) u toj molekuli. Formula molekule se uvijek sastoji od točno četiri znaka i ima sljedeći format:

- Prvi znak je oznaka prvog elementa (primjerice, C, O, N, itd.)
- Drugi znak je količina atoma prvog elementa (broj između 1 i 9)
- Treći znak je oznaka drugog elementa (primjerice, C, O, N, itd.)
- Četvrti znak je količina atoma drugog elementa (broj između 1 i 9)

Primjerice, ako funkcija primi formulu „H2O3“, treba ispisati 3; ako primi „O2N9“, treba ispisati 2; ako primi „C6H6“, treba ispisati 0. Pozovite funkciju iz main()-a.

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

void ispisi_kolicinu_kisika(string formula) {
    cout << "U formuli " << formula << " ima atoma kisika: ";

    if (formula[0] == 'O') {
        cout << formula[1] << endl;
    }
    else if (formula[2] == 'O') {
        cout << formula[3] << endl;
    }
    else {
        cout << '0' << endl;
    }
}

int main() {

    ispisi_kolicinu_kisika("H2O3");
    ispisi_kolicinu_kisika("O2N9");
    ispisi_kolicinu_kisika("C6H6");

    return 0;
}
```

Zadatak 4

Napišite funkciju `prikazi_sastav` koja od korisnika učitava formulu molekule i ispisuje pune nazive elemenata od kojih se molekula sastoji. Količine pojedinih elemenata u formuli mogu biti izostavljene ili napisane u obliku cijelog broja i uvijek ih treba zanemariti. U formuli se smiju pojaviti samo elementi:

- Au – zlato
- Se – selenij
- Ag – srebro
- Cl – klor
- Al – aluminij
- Si – silicij

Primjeri za testiranje:

- Ako funkcija učitava formulu „Au2Se3“, treba ispisati: zlato selenij
- Ako funkcija učitava formulu „AgCl“, treba ispisati: srebro klor
- Ako funkcija učitava formulu „AuCl3“, treba ispisati: zlato klor
- Ako funkcija učitava formulu „Al6Cl13Si2“, treba ispisati: aluminij klor silicij

Napišite i koristite dodatne funkcije prema želji. Pozovite funkciju `prikazi_sastav` iz `main()`-a.

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

bool je_li_broj(char c) {
    return c >= '0' && c <= '9';
}

string vrati_naziv_elementa(char z1, char z2) {
    if (z1 == 'A' && z2 == 'u') {
        return "zlato";
    }
    else if (z1 == 'S' && z2 == 'e') {
        return "selenij";
    }
    else if (z1 == 'A' && z2 == 'g') {
        return "srebro";
    }
    else if (z1 == 'C' && z2 == 'l') {
        return "klor";
    }
    else if (z1 == 'A' && z2 == 'l') {
        return "aluminij";
    }
    else if (z1 == 'S' && z2 == 'i') {
        return "silicij";
    }
    else {
        return "greska";
    }
}
```

```

void prikazi_sastav() {
    string formula;
    cout << "Upisite formulu: ";
    getline(cin, formula);

    // Uvijek gledamo i-ti znak i njegovog sljedbenika
    for (int i = 0; i < formula.length() - 1; i++) {

        if (!je_li_broj(formula[i])) {
            // Buduci da trenutni znak nije broj, nasao sam pocetak elementa
            cout << vrati_naziv_elementa(formula[i], formula[i + 1]) << " ";
            i++; // Rucno "premotamo" jos jedan znak jer smo ga obradili
        }
    }
}

int main() {
    prikazi_sastav();

    return 0;
}

```

Zadatak 5

Napišite funkciju koja od korisnika učitava cijeli broj između 1 i 5 (ako korisnik upiše neki drugi cijeli broj, ponavljajte učitavanje dok ne upiše broj između 1 i 5). Nakon ispravnog učitavanja, neka funkcija vrati učitani broj pozivatelju. U main()-u napravite polje od 5 elemenata i pomoću funkcije učitajte u njega brojeve između 1 i 5. Na kraju, ispišite sadržaj polja. Primjer rada programa:

```
Ucitavam 1. broj
Upisite broj: 4
Ucitavam 2. broj
Upisite broj: 5
Ucitavam 3. broj
Upisite broj: 3
Ucitavam 4. broj
Upisite broj: 9
Upisite broj: 12
Upisite broj: 1
Ucitavam 5. broj
Upisite broj: 6
Upisite broj: 5
Ispis brojeva: 4 5 3 1 5
```

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

int ucitaj() {
    int broj;
    do {
        cout << "Upisite broj: ";
        cin >> broj;
    } while (broj < 1 || broj > 5);

    return broj;
}

int main() {
    int brojevi[5];

    for (int i = 0; i < 5; i++) {
        cout << "Ucitavam " << (i + 1) << ". broj" << endl;
        brojevi[i] = ucitaj();
    }

    cout << "Ispis brojeva: ";
    for (int i = 0; i < 5; i++) {
        cout << brojevi[i] << " ";
    }
    cout << endl;

    return 0;
}
```

Zadatak 6

Napišite funkciju koja prima a = iznos ubrzanja (u m/s^2) i t = duljinu trajanja ubrzanja te vraća koliki put d je prošlo vozilo za to vrijeme. Koristite formulu za izračun puta: $d = \frac{1}{2}at^2$. Primjerice, ako vozilo 10 sekundi ubrzava 2.1 m/s^2 , onda će za to vrijeme proći 105 metara. Pozovite funkciju iz `main()`-a s istom akceleracijom i vremenima od 1, 10, 30 i 60 sekundi i ispišite rezultate.

Moguće rješenje:

```
#include <iostream>
#include <string>
#include <cmath>
using namespace std;

double put(double a, double t) {
    return (1.0 / 2.0) * a * pow(t, 2);
}

int main() {
    cout << "Put prijedjen za 1 s = " << put(2.1, 1.0) << " m" << endl;
    cout << "Put prijedjen za 10 s = " << put(2.1, 10.0) << " m" << endl;
    cout << "Put prijedjen za 30 s = " << put(2.1, 30.0) << " m" << endl;
    cout << "Put prijedjen za 60 s = " << put(2.1, 60.0) << " m" << endl;

    return 0;
}
```


Zadatak 7

Napišite program koji od korisnika učitava dva decimalna broja i operaciju (zbrajanje, oduzimanje, množenje ili dijeljenje). Ispišite rezultat tražene operacije koristeći po jednu funkciju za svaku od operacija.

Moguće rješenje:

```
#include <iostream>
using namespace std;

void zbroji(double a, double b) {
    cout << a << " + " << b << " = " << a + b << endl;
}

void oduzmi(double a, double b) {
    cout << a << " - " << b << " = " << a - b << endl;
}

void mnozi(double a, double b) {
    cout << a << " * " << b << " = " << a * b << endl;
}

void dijeli(double a, double b) {
    cout << a << " / " << b << " = " << a / b << endl;
}

int main() {
    double a, b;
    cout << "Upisite a: ";
    cin >> a;

    cout << "Upisite b: ";
    cin >> b;

    char op;
    cout << "Upisite operaciju: ";
    cin >> op;

    switch (op) {
        case '+':
            zbroji(a, b);
            break;
        case '-':
            oduzmi(a, b);
            break;
        case '*':
            mnozi(a, b);
            break;
        case '/':
            dijeli(a, b);
            break;
        default:
            cout << "Nepoznata operacija" << endl;
    }
    return 0;
}
```

Zadatak 8

UART je komunikacijski protokol koji se izvorno koristio za komunikaciju perifernih uređaja s računalom (primjerice, printer), a danas se uglavnom koristi na ugradbenim pločicama za komunikaciju mikroupravljača s ostalim komponentama (primjerice, modemom). UART paket izgleda ovako:

- 1. znak je uvijek '0'.
- Sljedećih 8 znakova su '0' ili '1' i predstavljaju poruku koju se želi prenijeti.
- Znak prije zadnjeg je paritetni znak koji se izračunava ovako: prebroje se sve jedinice u poruci i ako ih je paran broj, paritetni znak je '0', a ako ih je neparan broj, paritetni znak je onda '1'. Pošiljalac postavlja paritetni znak i primatelj ga ponovno provjerava. Ako postoji razlika, onda se dogodila pogreška u prijenosu.
- Zadnji znak je uvijek '1'.

Primjer poruke: 00100000101 – u primjeru vidimo da poruka sadrži 2 jedinice (paran broj jedinica), stoga je predzadnji, paritetni znak postavljen na '0'.

Napišite funkciju koja prima string s kompletnom primljenom porukom i vraća pozivatelju je li poruka ispravna provjerom prvog, zadnjeg i paritetnog znaka. Pozovite funkciju iz main()-a i ispišite je li ispravna ili neispravna.

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

bool poruka_ispravna(string poruka) {
    if (poruka[0] != '0') {
        return false; // Prvi znak nije '0'
    }

    int kolicina = 0;
    for (int i = 1; i < 9; i++) {
        if (poruka[i] == '1') {
            kolicina++;
        }
    }

    if (kolicina % 2 == 0 && poruka[9] == '1') {
        // Postoji parni broj jedinica u poruci, ali je paritetni znak 0
        return false;
    }
    else if (kolicina % 2 == 1 && poruka[9] == '0') {
        // Postoji neparan broj jedinica u poruci, ali je paritetni znak 1
        return false;
    }

    if (poruka[10] != '1') {
        return false; // Zadnji znak nije '1'
    }

    return true; // Nema gresaka
}

int main() {
```

```
if (poruka_ispravna("00100000101")) {  
    cout << "Poruka je ispravna" << endl;  
}  
else {  
    cout << "Poruka je neispravna" << endl;  
}  
return 0;  
}
```

Zadatak 9

Napišite funkciju koja prima string i ispisuje ga tako da prvo ispiše „>->-> “ , zatim znakove stringa razdvojene razmakom i na kraju „<-<-<“. Pozovite funkciju s nekim stringom.

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

void ispisi_naglaseno(string tekst) {
    cout << ">->-> ";
    for (int i = 0; i < tekst.length(); i++) {
        cout << tekst[i] << ' ';
    }
    cout << "<-<-<" << endl;
}

int main() {
    ispisi_naglaseno("nova knjiga!");

    return 0;
}
```

Zadatak 10

Napišite funkciju koja prima riječ i vraća je li palindrom. Palindrom je antički naziv za riječ koja čitana od početka prema kraju ili obrnuto glasi jednako (primjerice: kisik, ratar, potop). Pretpostavite da će riječ uvijek sadržavati mala slova bez ikakvih dodatnih znakova.

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

bool palindrom(string tekst) {
    int indeks_s_kraja = tekst.length() - 1;

    for (int i = 0; i < tekst.length() / 2; i++) {
        if (tekst[i] != tekst[indeks_s_kraja]) {
            return false;
        }
        indeks_s_kraja--;
    }

    return true;
}

int main() {

    cout << palindrom("nova knjiga!") << endl;
    cout << palindrom("kisik") << endl;
    cout << palindrom("brankova ulica") << endl;
    cout << palindrom("ratar") << endl;
    cout << palindrom("potop") << endl;
    cout << palindrom("racecar") << endl;
    cout << palindrom("saippuakivikauppias") << endl;
    cout << palindrom("hannah") << endl;

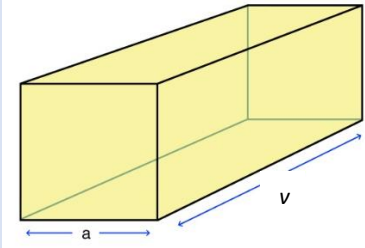
    return 0;
}
```

Zadatak 11

Napišite jednu funkciju koji prima sve potrebne parametre od pozivatelja te izračunava i vraća volumen odabrane pravilne prizme. Neka funkcija zna izračunati volumene sljedećih pravilnih prizmi:

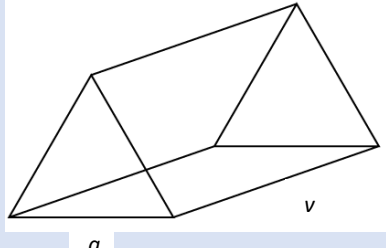
Kvadratna prizma ima volumen:

$$V = a^2 v$$



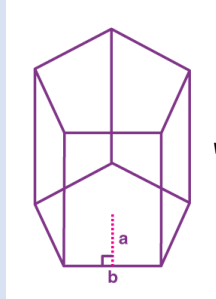
Trokutasta prizma ima volumen:

$$V = \frac{\sqrt{3}}{4} a^2 v$$



Pentagonalna prizma ima

$$\text{volumen: } V = \frac{5}{2} a b v$$



Moguće rješenje:

```
#include <iostream>
#include <cmath>
using namespace std;

double volumen(char vrsta_prizme, double v, double a, double b) {
    switch (vrsta_prizme) {
        case 'K':
            return pow(a, 2) * v;
        case 'T':
            return (sqrt(3) / 4) * pow(a, 2) * v;
        case 'P':
            return (5.0 / 2.0) * a * b * v;
    }
}

int main() {
    cout << volumen('K', 9.5, 4.5, 0.0) << endl;
    cout << volumen('T', 9.5, 4.5, 0.0) << endl;
    cout << volumen('P', 9.5, 4.5, 5.0) << endl;

    return 0;
}
```

Zadatak 12

Napišite program koja od korisnika učitava parni broj N i crta kvadrat stranice N (ako korisnik unese neparni broj, nemojte iscrtati ništa). Za crtanje treba koristiti '*' za jednu četvrtinu kvadrata, 'o' za drugu, '#' za treću i 'x' za četvrtu. Koristite funkcije gdje je zgodno kako biste pojednostavili izgled programa. Primjerice, za N=6 program, ispis može izgledati ovako:

```
***ooo
***ooo
***ooo
xxx###
xxx###
xxx###
```

Moguće rješenje:

```
#include <iostream>
using namespace std;

void iscrtaj_znak(int stupac, int n, int redak) {
    if (stupac <= n / 2 && redak <= n / 2) { // lijevo gore
        cout << '*';
    }
    else if (stupac <= n / 2 && redak > n / 2) { // lijevo dolje
        cout << 'x';
    }
    else if (stupac > n / 2 && redak <= n / 2) { // desno gore
        cout << 'o';
    }
    else { // desno dolje
        cout << '#';
    }
}

void iscrtaj_redak(int redak, int n) {
    for (int stupac = 1; stupac <= n; stupac++) {
        iscrtaj_znak(stupac, n, redak);
    }
    cout << endl;
}

void iscrtaj_kvadrat(int n) {
    for (int redak = 1; redak <= n; redak++) {
        iscrtaj_redak(redak, n);
    }
}

int main() {
    int n;
    cout << "Upisite stranicu: ";
    cin >> n;
    if (n % 2 == 0) {
        iscrtaj_kvadrat(n);
    }
    return 0;
}
```

Zadatak 13

Napišite funkciju koja prima cijeli broja a i vraća sljedeću vrijednost:

- Ako je a manji od 20, vratite 20.
- Ako je a veći od 30, vratite 30.
- U ostalim slučajevima vratite a .

U main()-u pripremite polje od 5 cijelih brojeva. Za svaki broj iz polja pozovite funkciju i ispišite vraćenu vrijednost.

Moguće rješenje:

```
#include <iostream>
using namespace std;

int ogranicici(int a) {
    if (a < 20) {
        return 20;
    }
    else if (a > 30) {
        return 30;
    }
    else {
        return a;
    }
}

int main() {

    int brojevi[5] = { -8, 523, 3, 24, 29 };

    for (int i = 0; i < 5; i++) {
        cout << ogranicici(brojevi[i]) << endl;
    }

    return 0;
}
```


Zadatak 14

Napišite funkciju koja simulira rad aparata za kavu. Neka funkcija primi ubačeni iznos novca i odabir kave (kratka kava, duga kava, cappuccino ili espresso, a pretpostavite da sve kave koštaju jednako i to 1 EUR) te ispiše koju je kavu pripremila i u kojim kovanicama je vratila ostatak novca. Neka funkcija uvijek vraća ostatak novca samo u kovanicama (2 EUR ili 1 EUR) i to u što manjoj količini. Primjerice, ako treba vratiti 5 EUR, onda funkcija uvijek vraća 2 EUR, 2 EUR, 1 EUR. Pazite na situaciju kad korisnik ubaci premalo novca.

Moguće rješenje:

```
#include <iostream>
#include <string>
#include <thread>
using namespace std;

string naziv_kave(int odabir) {
    switch (odabir) {
        case 1: return "kratka kava";
        case 2: return "duga kava";
        case 3: return "cappuccino";
        case 4: return "espresso";
    }
}

// Ova funkcija je napisana samo kao demonstracija kako se izvršavanje
// programa može pauzirati neko vrijeme. Ovo nije gradivo ispita i
// ne morate ga znati, pa umjesto ove funkcije mozete koristiti funkciju ispisi.
void ispisi_fancy(int odabir) {
    cout << "Pripremam: " << naziv_kave(odabir) << endl;

    for (int i = 0; i < 10; i++) {
        this_thread::sleep_for(100ms);
        cout << '.';
    }

    cout << "\n*BIIIIIP* Kava je skuhana, izvolite" << endl;
}

void ispisi(int odabir) {
    cout << "Pripremam: " << naziv_kave(odabir) << endl;
}

void vrati_novac(int ubaceni_novac) {
    cout << "\nPovrat novca:" << endl;
    while (ubaceni_novac > 0) {
        if (ubaceni_novac >= 2) {
            cout << "Vracam 2 EUR" << endl;
            ubaceni_novac -= 2;
        }
        else {
            cout << "Vracam 1 EUR" << endl;
            ubaceni_novac--;
        }
    }
}
```

```

}

void skuhaj_kavu(int odabir, int ubaceni_novac) {
    int cijena_kave = 1;
    if (ubaceni_novac < cijena_kave) {
        cout << "Nedovoljno sredstava" << endl;
        return;
    }

    ispisi_fancy(odabir);
    //ispisi(odabir);

    vrati_novac(ubaceni_novac - cijena_kave);
}

int main() {
    int odabir;
    cout << "Odaberite kavu: " << endl;
    cout << "1 = kratka kava" << endl;
    cout << "2 = duga kava" << endl;
    cout << "3 = cappuccino" << endl;
    cout << "4 = espresso" << endl;
    cout << "> ";
    cin >> odabir;

    int novac;
    cout << "Ubacite novac: ";
    cin >> novac;

    skuhaj_kavu(odabir, novac);

    return 0;
}

```

Zadatak 15

Napišite funkciju koja prima dva cijela broja i ispisuje sve brojeve od manjeg do većeg (pazite, ne znamo koji od dva primljena broja je manji, a koji je veći). Pozovite funkciju tri puta: u prvom pozivu neka je prvi parametar manji od drugog, u drugom pozivu neka je prvi parametar veći od drugog, a u trećem pozivu neka su im vrijednosti jednake.

Moguće rješenje:

```
#include <iostream>
using namespace std;

void ispisi(int a, int b) {
    // Ako je a < b, zamijenimo vrijednosti
    if (a > b) {
        int temp = a;
        a = b;
        b = temp;
    }

    for (int i = a; i <= b; i++) {
        cout << i << endl;
    }
}

int main() {

    ispisi(1, 10);
    cout << "-----" << endl;
    ispisi(10, 1);
    cout << "-----" << endl;
    ispisi(5, 5);

    return 0;
}
```

Zadatak 16

Napišite funkciju koja prima string sastavljen isključivo od malih slova. Neka funkcija sva uzastopna pojavljivanja nekog slova S ispiše skraćeno u obliku NS , gdje je N broj uzastopnog pojavljivanja slova S . Ako se slovo pojavi samo jednom, neka ga ispiše normalno. Primjerice, ako funkcija primi string SAAARRMMMMMA, treba ispisati: S3A2R5MA (jer imamo jedno S pa ga samo ispišemo, zatim imamo tri slova A pa pišemo 3A, dva slova R pa pišemo 2R, pet slova M pa pišemo 5M i jedno slovo A koje samo ispišemo).

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

void ispisi(string s) {
    char prethodni_znak = s[0];
    int broj_ponavljanja = 1;

    for (int i = 1; i < s.length(); i++) {
        if (s[i] == prethodni_znak) {
            // Nastavljamo niz
            broj_ponavljanja++;
        }
        else {
            // Niz je prekinut, ispisemo
            if (broj_ponavljanja == 1) {
                cout << prethodni_znak;
            }
            else {
                cout << broj_ponavljanja << prethodni_znak;
            }
            broj_ponavljanja = 1;
        }
        prethodni_znak = s[i];
    }

    // Na kraju ispisemo zadnje slovo
    if (broj_ponavljanja == 1) {
        cout << prethodni_znak;
    }
    else {
        cout << broj_ponavljanja << prethodni_znak;
    }
    cout << endl;
}

int main() {
    ispisi("SAAARRMMMMMA");
    ispisi("SARMA");
    ispisi("SSAARRMMAA");
    ispisi("SARRRRRRRRRRRRRRRRRRRMA");

    return 0;
}
```

Zadatak 17

Napišite funkciju koja prima string i znak i vraća koliko se puta u stringu pojavljuje taj znak. Pozovite funkciju iz main()-a i ispišite rezultat.

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

int prebroji(string s, char znak) {
    int n = 0;

    for (int i = 0; i < s.length(); i++) {
        if (s[i] == znak) {
            n++;
        }
    }

    return n;
}

int main() {

    string recenica = "a common mistake that people make when trying to design
something completely foolproof is to underestimate the ingenuity of complete fools";
    char znak = 'o';

    int kolicina = prebroji(recenica, znak);

    cout << "Pronadjeno znakova: " << kolicina << endl;

    return 0;
}
```

Zadatak 18

Napišite funkciju koja prima string koji predstavlja MAC adresu i vraća pozivatelju je li MAC adresa ispravna. MAC adresu čine 12 heksadecimalnih brojeva, a iza svaka dva broja dolazi crtica, osim iza zadnjeg. Pozovite funkciju s nekoliko ispravnih i neispravnih MAC adresa iz main()-a. Primjer ispravne MAC adrese: 00-B0-D0-63-C2-26

Moguće rješenje:

```
#include <iostream>
#include <string>
using namespace std;

bool heksadecimalni_broj(char c) {
    // Provjerimo je li c znamenka 0-9 ili slovo a-f ili A-F.
    return c >= '0' && c <= '9' ||
           c >= 'a' && c <= 'f' ||
           c >= 'A' && c <= 'F';
}

bool je_li_ispravna(string mac) {
    if (mac.length() != 17) {
        return false;
    }

    for (int i = 0; i < mac.length(); i++) {
        if (i == 2 || i == 5 || i == 8 || i == 11 || i == 14) {
            // Trazimo crticu
            if (mac[i] != '-') {
                return false;
            }
        }
        else {
            if (!heksadecimalni_broj(mac[i])) {
                return false;
            }
        }
    }

    return true;
}

int main() {

    cout << je_li_ispravna("00-B0-D0-63-C2-26") << endl;
    cout << je_li_ispravna("00-B0-D0-63-C2-2") << endl;
    cout << je_li_ispravna("FF-B1-D0-63-C2-26") << endl;
    cout << je_li_ispravna("00-G0-D0-63-C2-26") << endl;
    cout << je_li_ispravna("00-B0-D0-63-C2") << endl;
    cout << je_li_ispravna("50-81-40-FC-92-70") << endl;
    cout << je_li_ispravna("00-15-5D-12-FD-9C") << endl;
    cout << je_li_ispravna("0A-00-27-00-00-16") << endl;
    cout << je_li_ispravna("54-14-F3-FF-0E-E8") << endl;

    return 0;
}
```

Zadatak 19

Napišite jednu funkciju koja prima potrebne parametre te izračunava i vraća gustoću, masu ili volumen, prema korisnikovoj želji. Koristite sljedeće formule:

- Gustoća $\rho = \frac{m}{V}$
- Masa $m = \rho V$
- Volumen $V = \frac{m}{\rho}$

Pozovite funkciju tri puta kako biste dobili i ispisali prvo gustoću pa masu pa volumen.

Moguće rješenje:

```
#include <iostream>
using namespace std;

double izracunaj(char sto, double ro, double m, double v) {
    switch (sto) {
        case 'R': return m / v;
        case 'M': return ro * v;
        case 'V': return m / ro;
    }
}

int main() {

    cout << "Gustoca je " << izracunaj('R', 0.0, 2.75, 0.35) << " kg/m3" << endl;
    cout << "Masa je " << izracunaj('M', 7.857, 0.0, 0.35) << " kg" << endl;
    cout << "Volumen je " << izracunaj('V', 7.857, 2.75, 0.0) << " m3" << endl;

    return 0;
}
```

Zadatak 20

Napišite funkciju koja prima dva stringa jednake duljine i ispisuje tri retka:

- U prvom retku ispisuje znakove koji su u oba stringa nalaze na istom indeksu i jednaki su. Umjesto ostalih znakova ispisuju se podvlake.
- U drugom retku ispisuje znakove iz prvog stringa koji na istom indeksu u drugom stringu nemaju jednak znak. Umjesto ostalih znakova ispisuju se podvlake.
- U trećem retku ispisuje znakove iz drugog stringa koji na istom indeksu u prvom stringu nemaju jednak znak. Umjesto ostalih znakova ispisuju se podvlake.

Primjerice, ako su dva stringa „easefulnesses“ i „easygoingness“, funkcija treba ispisati:

```
eas__n__s
__eful_esse_
__ygoi_gnes_
```

Moguće rješenje:

```
#include <iostream>
using namespace std;

void ispisi(string a, string b) {
    // Prvi redak
    for (int i = 0; i < a.length(); i++) {
        if (a[i] == b[i]) {
            cout << a[i];
        }
        else {
            cout << "_";
        }
    }
    cout << endl;

    // Drugi redak
    for (int i = 0; i < a.length(); i++) {
        if (a[i] != b[i]) {
            cout << a[i];
        }
        else {
            cout << "_";
        }
    }
    cout << endl;

    // Treći redak
    for (int i = 0; i < a.length(); i++) {
        if (a[i] != b[i]) {
            cout << b[i];
        }
        else {
            cout << "_";
        }
    }
    cout << endl;
}
```



```
int main() {  
    ispisi("easefulnesses", "easygoingness");  
    cout << endl;  
    ispisi("branka", "krafna");  
    cout << endl;  
    ispisi("funny", "funny");  
    cout << endl;  
    return 0;  
}
```