***DO WHILE, WHILE, IF ELSE***

***IF***

***Zadatak 3: Broj u intervalu***

Napisati program koji učitava ceo broj ***n***, a zatim ispisuje poruku da li je taj broj između 1 i 100, ali ne i između 10 i 20.

Primer. Ako je **n**=50, program ispisuje „Uneseni broj je u zadatom skupu.“. Ako je **n**=-50 ili **n**=15, ispisuje se „Uneseni broj nije u zadatom skupu.“.

int x;

if(1 < x && x < 100 && !(10 < x && x < 20))

printf("Uneseni broj je u zadatom skupu.\n");

else

printf("Uneseni broj nije u zadatom skupu.\n");

***Zadatak 4: Cifre***

Napisati program koji za uneti znak proverava i ispisuje da li je on cifra.

char karakter;

if(karakter - '0' >= 0 && karakter - '0' <= 9)

{

printf("Karakter je cifra.");

}

else

{

printf("Karakter nije cifra.");

}

***Zadatak 5: Stranica kvadrata***

Napisati program koji za unetu vrednost površine kvadrata proverava da li je ona pozitivan broj, i ako jeste izračunava i ispisuje kolika je stranica tog kvadrata, a u suprotnom ispisuje odgovarajuću poruku.

float a, P;

if (P >= 0)

{

a = sqrt(P);

printf("Stranica tog kvadrata je %f.\n", a);

}

else

printf("Povrsina ne moze biti negativna!\n");

***Zadatak 7: Slova***

Napisati program koji za uneti znak proverava i ispisuje da li je on slovo.

char c;

if('a' <= c && c <= 'z' || 'A' <= c && c <= 'Z')

printf("Uneseni znak jeste slovo.\n");

else

printf("Uneseni znak nije slovo.\n");

***Zadatak 8: Kupovina***

Napisati program koji učitava cenu proizvoda ***n***, a zatim ispisuje poruku da li korisnik kupuje taj proizvod ili ne. Korisnik kupuje proizvod ukolikoje cena tog proizvoda između definisane donje i gornje granice.

const float DONJA\_GRANICA = 1000.0;

const float GORNJA\_GRANICA = 2500.0;

float cena;

printf("Proizvod koji kosta %f dinara ", cena);

if(cena >= DONJA\_GRANICA && cena <= GORNJA\_GRANICA)

printf("je kupljen.\n");

else

printf("nije kupljen.\n");

***IF ELSE***

***Zadatak 2: Broj cifara***

Napisati program koji za unet prirodan broj **n** ispisuje poruku da li taj broj ima jednu, dve, tri ili više od tri cifre.

int n;

if(n <= 9)

printf("Uneti broj ima 1 cifru.");

else if(n >= 10 && n <= 99)

printf("Uneti broj ima 2 cifre.");

else if(n >= 100 && n <= 999)

printf("Uneti broj ima 3 cifre.");

else

printf("Uneti broj ima vise od 3 cifre.");

***Zadatak 3: Intervali slova***

Napisati program koji za uneto malo slovo ispisuje da li se ono nalazi u intervalu od *a* do *k* ili u intervalu od *l* do *z*. Ako nije uneto malo slovo ispisati odgovarajuću poruku.

char c;

if('a' <= c && c <= 'k')

printf("Uneto slovo je u intervalu izmedju a i k.");

else if('l' <= c && c <= 'z')

printf("Uneto slovo je u intervalu izmedju l i z.");

else

printf("Nije uneto malo slovo.");

***Zadatak 5: Računanjeproseka***

Napisati program koji računa i ispisuje prosek na kraju školske godine.Korisnik unosi ocene iz predmeta Matematika, Srpski jezik, Fizičko vaspitanje i Informatika, a potom program ispisuje prosek učenika (ukoliko učenik ima makar jednu jedinicu, prosek je automatski 1.00).

int oMat, oSJ, oFV, oInf;

float prosek;

if(oMat == 1 || oSJ == 1 || oFV == 1 || oInf == 1){

prosek = 1;

}

else {

prosek = (oMat + oSJ + oFV + oInf) / 4.0;

}

printf("Vas prosek je %.2f", prosek);

***Zadatak 6: Broj pozitivnih***

Napisati program koji učitava četiri cela broja **a**, **b**, **c** i **d**, a zatim izračunava i prikazuje koliko među njima ima pozitivnih.

int a, b, c, d, brPoz = 0;

if(a > 0)

brPoz++;

if(b > 0)

brPoz++;

if(c > 0)

brPoz++;

if(d > 0)

brPoz++;

printf("Medju unetim brojevima je %d pozitivnih", brPoz);

***Zadatak 7: Plaćanje računa***

Napisati program koji simulira plaćanje računa. Korisnik unosi iznos računa, kao i dan u mesecu kada izvršava plaćanje računa. Ukoliko korisnik plaća račun nakon 20. dana u mesecu, ispisuje se poruka da mora da plati iznos koji je za 10% veći u odnosu na uneti iznos (ispisati novi iznos koji korisnik mora da plati). Ukoliko plaća pre tog dana ili taj dan, proverava se da li je iznos računa veći ili jednak od 1000 dinara. Ukoliko jeste, korisnik dobija poruku da je uspešno platio račun i da je dobio status ZLATNOG POTROŠAČA. Ukoliko je taj iznos pak manji od 1000 dinara, dobija poruku samo da je uspešno platio račun.

unsigned dan;

float iznos;

if(dan > 20){

printf("Prosao je 20 dan u mesecu, morate da platite %.2f dinara.", iznos \* 1.1); //1.1 predstavlja 110%

}

else if (iznos >= 1000){

printf ("Racun uspesno placen. Dobili ste status ZLATNOG POTROSACA.");

}

else{

printf ("Racun uspesno placen.");

}

***Zadatak 8: Obim i površina***

Napisati program koji korisniku nudi mogućnost da izabere jednu od tri figure: 1) krug, 2) pravougaonik ili 3) trougao, i šta želi da izračuna: p) površinu ili o) obim, a zatim izračunava traženu veličinu. Površina trougla računa se Heronovom formulom: ako je s=(a+b+c)/2, površina je P=√𝑠(𝑠−𝑎)(𝑠−𝑏)(𝑠−𝑐)

const float PI = 3.14;

float a, b, c, r, s, O, P;

char izbor1, izbor2;

if(izbor1 == 'k') {

printf("Uneti poluprecnik kruga: ");

if(izbor2 == 'o')

printf("Obim kruga je %f", 2 \* r \* PI);

else if(izbor2 == 'p')

printf("Povrsina kruga je %f", r \* r \* PI);

}

else if(izbor1 == 'p'){

if(izbor2 == 'o')

printf("Obim pravougaonika je %f", 2 \* (a + b));

else if(izbor2 == 'p')

printf("Povrsina pravougaonika je %f", a \* b);

}

else if(izbor1 == 't'){

if(izbor2 == 'o')

printf("Obim trougla je %f", a + b + c);

else if(izbor2 == 'p'){

s = (a + b + c) / 2;

printf("Povrsina trougla je %f", sqrt(s \* (s - a) \* (s - b) \* (s - c)));

}

}

***WHILE***

***Zadatak 2: Štednja***

Napisati program koji učitava sumu novca **suma** koju ulažemo u štednju, sumu novca **cilj** koju želimo da dobijemo i mesečnu kamatu banke **kamata**, a zatim izračunava koliko meseci će biti potrebno da bi se dostigla ciljna suma. Primer. Ako je **suma** = 1000,0 , **cilj** = 1200,0 i **kamata** = 5,0 %, rezultat je 4: nakon jednog meseca suma u banci je 1000,0 **∙** 1,05 = 1050,0 dinara, nakon drugog 1050,0 **∙** 1,05 = 1102,5, nakon trećeg 1102, 5 **∙** 1,05 = 1163,14, a nakon četvrtog 1221,29, čime je dostignuta ciljna suma.

float suma, kamata, cilj;

int meseci = 0;

while(suma < cilj) {

meseci++;

suma = suma \* (1 + kamata \* 0.01); //ili suma \*= 1 + kamata \* 0.01;

}

printf("Do ciljne sume cete stici nakon %d meseci.\n", meseci);

***Zadatak 3: Zbir cifara***

Napisati program koji prikazuje zbir cifara unetog celog broja **n**.

Primer. Ako je unet broj **n** = 1234, rezultat je 10. Ako je unet broj **n** = 8, rezultat je 8.

int n, zbir = 0;

while(n != 0){

zbir += n % 10;

n /= 10;

}

printf("Zbir cifara je: %i\n", zbir);

***Zadatak 4: Obrnute cifre***

Napisati program koji za unetu vrednost prirodnog broja **n** ispisuje broj koji ima iste cifre kao uneti broj, ali u obrnutom redosledu.

Primer. Ako je unet broj **n** = 1234, ispisuje se: „Broj sa obrnutim redosledom cifara je 4321.”

int n, br = 0;

while(n != 0) {

br = br \* 10 + n % 10; //nije isto sto i br \*= 10 + n % 10;

n /= 10;

}

printf("Broj sa obrnutim redosledom cifara je %i.\n", br);

***Zadatak 5: Rastojanja tačaka***

Napisati program koji učitava parove tačaka u koordinatnoj ravni sve dok rastojanje neke dve uzastopne tačke ne bude veće od 10. Tačke se učitavaju u obliku koordinata **x** i **y**, a rastojanje tačke (**x**, **y**) od prethodne tačke (**xPret**, **yPret**) se računa pomoću Pitagorine teoreme: √(𝑥−𝑥𝑃𝑟𝑒𝑡)2+(𝑦−𝑦𝑃𝑟𝑒𝑡) 2. Na kraju se ispisuje koliko tačaka je učitano.

Primer. Ako se unose tačke (0, 0), (7, 7), (-3, 7), (-10, -1), unos se prekida i ispisuje se: “Ucitano je 4 tacaka.”

float x, y, xPret, yPret;

int brojac = 2;

while(sqrt(pow(x - xPret, 2) + pow(y - yPret, 2)) <= 10) {

xPret = x;

yPret = y;

brojac++;

}

printf("\nUcitano je %i tacaka.\n", brojac);

***Zadatak 6: Novčanice***

Napisati program koji učitava sumu novca **suma**, broj novčanica **n** koje ima na raspolaganju (od najveće ka najmanjoj), a zatim te novčanice (od najveće ka najmanjoj), i ispisuje koliko je najmanje novčanica potrebno da se isplati tačna suma.

Napomene. Od svake učitane novčanice imamo na raspolaganju dovoljnu količinu. Smatrati da je unesena suma koja se može tačno isplatiti i da je najmanje novčanica potrebno ako se koriste što krupnije novčanice. Unos novčanica se prekida ako je sumu moguće isplatiti pomoću već unesenih novčanica.

Primeri. 1) Ako je **suma** = 1230, **n** = 4 i unose se novčanice od 1000, 100, 20 i 10 dinara, rezultat je 5: 1230 = 1000 + 2 ∙ 100 + 20 + 10; 2) Ako je **suma** = 12345**, n** = 7 i unose se novčanice od 10000, 200, 50, 5, unos se tu prekida, jer su učitane novčanice dovoljne za isplatu, a rezultat je 23: 12345 = 10000 + 11 ∙ 200 + 2 ∙ 50 + 9 ∙ 5.

int n, suma, novc, kol, i = 1, brojac = 0;

while(i <= n && suma > 0) {

kol = suma / novc;

suma = suma % novc; //moze i suma %= novc;

brojac += kol;

i++;

}

printf("Za isplatu je korisceno %d novcanica.\n", brojac);

***Zadatak 8: Niz cifara***

Niz cifara je zadat na sledeći način: prva četiri elementa su 2, 0, 1, 8, a svaki sledeći se dobija kao poslednja cifra zbira prethodna četiri. (Npr. peti član niza je 1 jer je 2 + 0 + 1 + 8 = 11, šesti je 0 jer je 0 + 1 + 8 + 1 = 10, itd). Napisati program koji izračunava počevši od koje pozicije u nizu se prvi put pojavljuje četvorka 2, 3, 8, 3.

Primer. Program ispisuje: „Cetvorka 2, 3, 8, 3 se pojavljuje od 642. pozicije.”

int c1 = 2, c2 = 0, c3 = 1, c4 = 8, c5, brojac = 1;

while (!(c1 == 2 && c2 == 3 && c3 == 8 && c4 == 3)) {

c5 = (c1 + c2 + c3 + c4) % 10;

c1 = c2;

c2 = c3;

c3 = c4;

c4 = c5;

brojac++;

}

printf("Cetvorka 2, 3, 8, 3 se pojavljuje od %i. pozicije.\n", brojac);

***Zadatak 9: NZD***

Napisati program koji Euklidovim algoritmom određuje najveći zajednički delilac (NZD) dva zadata prirodna broja **a** i **b**.

Uputstvo. (1) NZD(**a**,**b**) se računa, polazeći od brojeva **a** i **b**, uzastopnim oduzimanjem manjeg broja od većeg, sve dok se ne dobije nula. Poslednja razlika različita od nule je traženi NZD. (2) Postupak se može značajno ubrzati ako se umesto oduzimanja svaki put računa ostatak pri deljenju.

Primer. Neka su uneseni brojevi **a** = 14 i **b** = 49. (1) Oduzimajući od većeg broja manji dobijamo: 49 - 14 = 35, 35 – 14 = 21, 21 – 14 = 7, 14 – 7 = 7, 7 – 7 = 0. Poslednji broj različit od nule je 7. (2) Računanjem ostataka pri deljenju dobijamo: 49 % 14 = 7, 14 % 7 = 0, i opet je rezultat 7.

|  |  |
| --- | --- |
| int a, b, pom;  while(a != 0) {  if (a < b) {  pom = a;  a = b;  b = pom;  }  a -= b;  }  printf("Najveci zajednicki delitelj je: %i\n", b); | int a, b, pom;  if(a < b) {  pom = a;  a = b;  b = pom;  }  while(c != 0) {  //nema potrebe da svaki put proveravamo da li je a>=b jer je b dobijen kao ostatak pri deljenju s a  c = a % b;  a = b;  b = c;  } |

***Zadatak 10: Prost broj***

Napisati program koji proverava da li je uneti prirodan broj **n** prost.

Uputstvo. Treba proveriti da li je **n** deljiv nekim od brojeva većim od 1, a manjim od **n**. Dovoljno je proveravati samo brojeve manje ili jednake od √𝑛.

Primer. Ako je unesen broj **n** = 11, proverava se da on nije deljiv sa 2 ni sa 3 (√11=3,32) i ispisuje se: „Broj je prost”. Ako je unesen broj **n** = 14, on je deljiv sa 2, pa se ispisuje: „Broj nije prost“.

|  |  |
| --- | --- |
| int n, koren, br = 2, prost = 1;  koren = sqrt(n);  while(prost && br <= koren) {  if (n % br == 0)  prost = 0;  br++;  }  if (prost)  printf("Broj je prost\n");  else  printf("Broj nije prost\n"); | int n, br = 2, prost = 1;  while(prost && br \* br <= n) {  if (n % br == 0)  prost = 0;  br++;  }  if (prost)  printf("Broj je prost\n");  else  printf("Broj nije prost\n"); |

***DO WHILE***

***Zadatak 3: Najstarija osoba***

Napisati program koji učitava godine rođenja svih korisnika gerontološkog centra. Korisnik unosi ove godine jednu po jednu. Na kraju korisnik unosi nulu, nakon koje program ispisuje koliko godina ima najstarija osoba u tom gerontološkom centru. Tekuću godinu definisati kao celobrojnu konstantu.

Primer. Ako je trenutna godina 2018 i unose se godine: 1956, 1948, 1950, 1956, 1949, učitavanje se prekida nakon unosa broja 0, ispisuje se: „Najstarija osoba ima 70 godina.” (jer je 70 = 2018 - 1948)

const int TEKUCA\_GODINA = 2018;

int godina, minGodina = -1;

do {

if(minGodina == -1 || (godina < minGodina && godina != 0))

minGodina = godina;

}

while(godina != 0);

printf("Najstarija osoba ima %d godina.\n", TEKUCA\_GODINA - minGodina);

***Zadatak 4: Čokolade***

Napisati program koji učitava količinu para koju kupac ima i cene čokolada koje on želi da kupi. Kada ukupna cena čokolada bude veća od količine para koje kupac ima, prekinuti učitavanje cena i ispisati koliko čokolada kupac može da kupi.

Primer. Ako kupac ima 1000 dinara i unose se cene: 130, 280, 90, 280, 100, 150, učitavanje se prekida nakon unosa broja 150 (jer je 130 + 280 + 90 + 280 + 100 + 150 > 1000) i ispisuje se: „Kupac moze da kupi 5 cokolada.”

int novac, cena, broj = 0, suma = 0;

do {

suma += cena;

broj++;

}

while(suma <= novac);

printf("Kupac moze da kupi %d cokolada.\n", broj - 1);

***Zadatak 5: Zbir cifara***

Napisati program koji učitava broj koji ima tri ili više cifara i prikazuje zbir njegovih cifara. Primer. Ako korisnik unese broj 64 ispisuje se poruka o grešci i unos se ponavlja. Ako se unese broj 6416 ispisuje se: „Zbir cifara je 17.”

int n, zbir = 0;

do {

zbir += n % 10;

n /= 10;

}

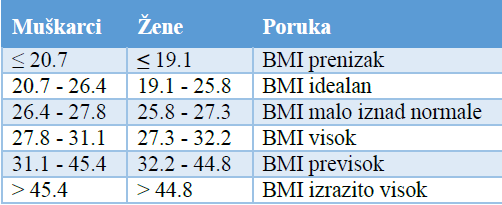
while(n != 0);

printf("Zbir cifara je %d.\n", zbir);

***Zadatak 6: Indeks telesne mase***

Napisati program koji na osnovu učitane težine (kg), visine (m) i pola (m / z) osobe izračunava njegov indeks telesne mase (BMI indeks) i ispisuje ga uz odgovarajuću poruku. U slučaju greške pri unosu, program treba da obezbedi ponovni unos vrednosti sve dok korisnik ne unese ispravne vrednosti za svaku od promenljivih.

Napomena. BMI se izračunava po sledećoj formuli *BMI = težina (kg) / (visina (m)* ***∙*** *visina (m))*. Poruke koje se ispusuju zavise od BMI indeksa i pola osobe, a one su prikazane u sledećoj tabeli:



Primer. Ako korisnik unese težinu 100 kg, visinu 1.82 m i pol m, ispisuje se: „BMI visok (BMI = 30.2)”

float tezina, visina, bmi;

char pol;

do {

if(tezina <= 0)

printf("Tezina mora biti pozitivan broj!\n");

}

while(tezina <= 0);

do {

if(visina <= 0)

printf("Visina mora biti pozitivan broj!\n");

}

while(visina <= 0);

do {

if(!(pol == 'm' || pol == 'z')) //uslov moze ovako

printf("Morate uneti m ili z!\n");

}

while(pol != 'm' && pol != 'z'); //a moze i ovako

bmi = tezina / (visina \* visina);

if (pol == 'm' && bmi <= 20.7 || pol == 'z' && bmi <= 19.1) {

printf("BMI prenizak");

}

else if(pol == 'm' && bmi > 20.7 && bmi <= 26.4 || pol == 'z' && bmi > 19.1 && bmi <= 25.8) {

printf("BMI idealan");

}

else if(pol == 'm' && bmi > 26.4 && bmi <= 27.8 || pol == 'z' && bmi > 25.8 && bmi <= 27.3) {

printf("BMI malo iznad normale");

}

else if(pol == 'm' && bmi > 27.8 && bmi <= 31.1 || pol == 'z' && bmi > 27.3 && bmi <= 32.2) {

printf("BMI visok");

}

else if(pol == 'm' && bmi > 31.1 && bmi <= 45.4 || pol == 'z' && bmi > 32.2 && bmi <= 44.8) {

printf("BMI previsok");

}

else {

printf("BMI izrazito visok");

}

printf(" (BMI = %.1f)\n", bmi);

***Zadatak 9: Slučajan broj***

Napisati program koji generiše slučajan broj od 1 do 100 i omogućava korisniku pogađanje broja uz prikaz odgovarajućih poruka u zavisnosti od toga da li je pogodio, da li je uneo manji ili je uneo veći broj. Nakon pogođenog broja, ispisati broj pokušaja.

Napomena. Slučajan broj generiše se funkcijom rand. On može biti u intervalu od 0 do RAND\_MAX (I rand i RAND\_MAX)

Primer. Neka je program „zamislio” broj 64. Ako korisnik unese broj 50, ispisuje se „Uneli ste premali broj”. Ako korisnik zatim unese broj 75, ispisuje se „Uneli ste preveliki broj”. Konačno, ako korisnik unese broj 64, ispisuje se „Pogodili ste iz 3 pokusaja”.

int n, br, brPokusaja = 0;

// srand(time(0)); //postavljanje semena randoma

br = rand() % 100 + 1;

do {

if(n > br)

printf("Uneli ste prevelik broj!\n\n");

else if(n < br)

printf("Uneli ste premali broj!\n\n");

brPokusaja++;

}

while(n != br);

printf("Pogodili ste iz %d pokusaja\n", brPokusaja);