# Zadaci – 12. dio, *Dodatak*

Zadatak 126:

Napravite program koji će provjeravati da li je uneseni broj n prost broj. Koristite:

1. *while-*petlju
2. *do-while*-petlju

Za pomoć pogledajte ponovo zadatak 99.

*Rješenje pomoću do-while-petlje se nalaze na stranici 160.*

Zadatak 127:

Optimizirajte prethodni program *(nije bitno u kojoj vrsti petlje je riječ)*:

1. Kao što vidite, zadatak smo riješili tako što smo brojali sa koliko je brojeva djeljiv broj **n**. Međutim, program ne nudi optimalno rješenje. Moguće je smanjiti korištenje resursa za dva puta, tako što ćemo provjeravati djeljivost broja **n** samo sa brojevima iz intervala od **2** do **n/2.** Primjer:

Za n=36597, provjeravat ćemo djeljivost broja **n** samo u intervalu od 2 do 18298, jer broj 36597 nije sigurno djeljiv ni sa jednim brojem iz intervala 18299 do 36596.

1. Dodatno optimiziranje programa:

Ako smo u toku provjeravanja (izvršavanja petlje) našli da je broj **n** djeljiv sa jednim brojem (brojem većim od 1) možemo odmah prekinuti petlju - jer broj **n** sigurno nije prost.

Poslije petlje slijedi *if-else*-iskaz (dvostruki izbor) koji treba, na osnovu nekog uslova, ispisati "…je prost" ili "…nije prost".

Pošto, *if*-*else*-iskaz 'ne zna' da li se petlja prekinula, moramo prethodno u *if*-iskazu (n%i==0) promijeniti vrijednost neke varijable tako da *if-else*-iskaz koji ispisuje poruku može 'znati' da li se *if*-iskaz (n%i==0) bio istinit ili neistinit izraz.

Promjena neke vrijednosti u *if*-iskazu (n%i==0) može da bude: brojac=1 ili prost=false ili sl.

(*Napomena*: varijable brojac ili prost moraju da imaju neku početnu vrijednost prije izvršavanja petlje. Ta početna vrijednost mora biti različita od vrijednosti *1* za varijablu brojac i različita od vrijednosti *false* za varijablu prost.)

Prekidanje petlje možete vršiti na dva načina:

1. Pomoću dodatnog uslova za prekid petlje.

Ako koristite varijablu brojac, uslov bi glasio:

* + i<=n/2 && brojac!=1 (čita se: ponavlja ako je …)

*dovoljno je da jedan uslov* ***ne*** *bude zadovoljen i petlja se prekida*

* + !(i>n/2 || brojac==1) (čita se: ponavlja ako je …)

*dovoljno je da jedan uslov bude zadovoljen i petlja se prekida*

*(*Ako su vam ovi logički uslovi nejasni pročitajte ponovo *Workshop – 6. dio - logički operatori*!)

1. Pomoću naredbe break: Kada se prvi put u petlji izvrši naredba break, petlja se prekida i nastavlja se sa izvršavanjem naredne linije koda koja slijedi nakon petlje.

*Rješenje pomoću while-petlje i varijable* prost *sa oba načina prekida petlje se nalazi na str. 160.*

Zadatak 128:

Napraviti program koji će ispisati sve trocifrene brojeve koji su djeljivi sa sumom svojih cifara, te utvrditi koliko ukupno ima takvih brojeva.

Program riješiti u skladu sa fazama programiranja!

*Rješenje se nalaze na stranici 162.*

*Pomoć:*

* postavit ćemo vrijednost za varijablu brojac na 0
* ispitat ćemo sve brojeve od 100 do 999, koristeći brojač petlje i
* izračunat ćemo *stotice* tako što ćemo podijeliti trenutni broj koji ispitujemo (broj i) sa 100 i odbaciti decimalnu vrijednost

(npr. **4**67:100 = 4.67, int(4.67) = **4**)

* izračunat ćemo *desetice* tako što ćemo, pomoću operatora za ostatak dijeljenja %, modularno dijeliti trenutni broj (i) sa 100 i dobit ćemo zadnje dvije cifre, a taj rezultat ćemo cijelo-brojno podijeliti sa 10, kao što smo to činili u prethodnom koraku

(npr. 4**6**7%100 = 67, 67/10 = 6.7, int(6.7) = **6**)

* izračunat ćemo *jedinice* tako što ćemo trenutni broj (i) modularno dijeliti sa 10

(npr. 46**7**%10 = **7**)

* izračunat ćemo sumu cifara trenutnog broja (i)
* ako je trenutni broj (i) djeljiv sa sumom svojih cifara ispisat ćemo ga na ekran i povećat ćemo brojac za 1
* poslije petlje ispisat ćemo vrijednost varijable brojac

Program treba ispisati ukupno 180 trocifrenih brojeva koji su djeljivi sa sumom svojih cifara.

**Zadatak 129:**

Riješite prethodni program pomoću tri ugniježđene *for*-petlje! Nacrtajte dijagram toka i na osnovu njega napišite C++ kôd.

*Rješenje se nalaze na stranici 163.*

Zadatak 130:

Napravite program u kome ćete inicijalizirati dvije varijable (a, b) tipa *integer*, a zatim te vrijednosti varijabli zamijeniti i ispisati na ekran prvo u hexadecimalnom, pa u oktalnom, a zatim u decimalnom brojnom sistemu.

*Rješenje se nalaze na stranici 163.*

Zadatak 131:

Napravite program u kome će te deklarisati tri varijable (a, b i c) tipa *integer*.

* varijabli a dodijelite neku vrijednost hexadecimalnom obliku
* varijabli b dodijelite neku vrijednost oktalnom obliku
* varijabli c dodijelite neku vrijednost decimalnom obliku
* ispišite vrijednost varijabli a, b i c u decimalnom obliku

*Rješenje se nalaze na stranici 163.*

**Zadatak 132:**  *(zadatak sa ispita: UPR - 29.01.2005.)*

Izraditi program u programskom jeziku C++ koji izračunava izraz:



Povratna vrijednost funkcije ZbirKubova i parametri **m** i **n** su tipa *int*. Zadatak riješiti u skladu sa fazama programiranja.

Omogućiti ponavljanje unosa ulaznih podataka (čitavog programa) ako korisnik to želi.

*Rješenje se nalaze na stranici 164.*

Zadatak 133:

U zadatku 149 je korištenja funkcija ZbirKubova sa povratnom vrijednošću i sa pozivom *pass-by-value*. Prepravite funkciju u ZbirKubova2, tako da funkcija nema povratnu vrijednost i da za izlaznu vrijednost iz funkcije koristite referencu (*pass-by-reference*).

Objasnite razliku između ovog programa i program iz zadatka 149.

*Riješen kôd se nalazi na stranici 166.*

*Pomoć:*

* u oba zadatka (149 i 150) ispis na ekran treba da se vrši u glavnom programu
* izlaz iz funkcije ZbirKubova (zadatak 149) je proslijeđen pomoću naredbe return
* izlaz iz funkcije ZbirKubova2 treba da se smjesti u referentnu varijablu
* u funkciji ZbirKubova2 nećemo koristiti naredbu return, s toga će ona biti tipa void
* u funkciji ZbirKubova2 koristit ćemo tri parametra: treći parametar će biti referentna varijabla (izlazni parametar)

**Zadatak 134:**  *(zadatak sa ispita: UPR - 16.04.2005.)*

Napišite program sa izbornikom (menî) za izračunavanje površine geometrijskih likova: kvadrat, pravougaonik i krug. Za izračunavanje površine koristiti funkcije. Menî treba da sadrži i izbor za kraj izvršenja programa.

*Riješen kôd se nalazi na stranici 167.*

Rješenje:

|  |  |
| --- | --- |
| Rješenje zadatka br. **126** *(pomoću while-petlja)*: | |
| 4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24:  25: | void main()  {  int n, brojac;  cout << "Unesi broj: ";  cin >> n;  brojac = 0;    int i=2;  while (i<=n/2)  {  if (n%i == 0)  brojac++;  i++;  }  if (brojac == 0)  cout << "Broj je prost\n";  else  cout << "Broj nije prost\n";  } |

|  |  |
| --- | --- |
| Rješenje zadatka br. **127 a** *(optimiziran program uz pomoć dodatnog uslova za prekid)*: | |
| 4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24:  25:  26: | void main()  {  int n;  bool prost;  cout << "Unesi broj: ";  cin >> n;  prost = true; // broj je prost (nevin) dok mu se ne dokaže da je složen (kriv)  int i=2;  while (i<=n/2 && prost==true)  {  if (n%i == 0)  prost=false; // dokazano da je broj slozen – dokazana je krivica  i++;  }  if (prost) // if (prost ==false)  cout << "Broj je prost\n";  else  cout << "Broj nije prost\n";  } |

|  |  |
| --- | --- |
| Rješenje zadatka br. **127 b** *(optimiziran program uz pomoć naredbe break)*: | |
| 4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24:  25:  26:  27:  28:  29: | void main()  {  int n;  bool prost;  cout << "Unesi broj: ";  cin >> n;  prost = true;  int i=2;  while (i<=n/2)  {  if (n%i == 0)  {  prost=false; // ova se naredba mora prije prekida izvrsiti  break; // prekid - izlaz iz petlje  }  i++;  }  if (prost) //ili if (prost ==false)  cout << "Broj je prost\n";  else  cout << "Broj nije prost\n";  } |

|  |
| --- |
| Rješenje zadatka br. **128** *(u skladu sa fazama programiranja)*: |
| **1. Analiza i specifikacija:**   |  |  | | --- | --- | | **Input** | **Output** | | - | svi trocifreni brojevi djeljivi sa sumom svojih cifara | | brojač ispisanih brojeva |   Ograničenja: *nema*  **2. Dizajn**   |  |  |  |  | | --- | --- | --- | --- | | **Podatkovni objekt** | **Vrsta vrijednosti** | **Tip objekta** | **Ime objekta** | | brojač ispisanih brojeva | varijabla | int | brojac | | stotice | varijabla | int | s | | desetice | varijabla | int | d | | jedinice | varijabla | int | j | | suma cifara | varijabla | int | suma\_cifara |      |  |  |  | | --- | --- | --- | | **3. Algoritam – blok dijagram** | **4. Kôd (code)** | | | 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24: | #include <iostream>  using namespace std;  void main()  {  int brojac, s, d, j, suma\_cifara;  brojac = 0;  for (int i=100; i<=999; i++)  {  s = i / 100; // int/int = int  d = (i % 100) / 10;  j = (i % 10);    suma\_cifara = s + d + j;  if (i%suma\_cifara == 0)  {  brojac++;  cout << i << " ";  }  }  cout << "\nUkupno: " << brojac << endl;  } | |

|  |  |
| --- | --- |
| Rješenje zadatka br. **129**: | |
| 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21: | void main()  {  int brojac;  brojac = 0;  for (int s=1; s<=9; s++)  {  for (int d=1; d<=9; d++)  {  for (int j=1; j<=9; j++)  {  if ((s\*100+d\*10+j)%(s+d+j)==0)  {  brojac++;  cout <<s<<d << j << " ";  }  }  }  }  cout << "\nUkupno: " << brojac << endl;  } |

*Nakon što riješite zadatak, pročitajte sljedeću napomenu:*

Ovaj program ima jednu sitnu grešku. Pokušajte je otkriti. Usporedite ispis ovog programa s ispisom prethodnog programa. Ovaj program će ispisati ukupno 108 brojeva, dok će prethodni program ispisati ispravan rezultat, tj. 180.

*Rješenje ove 'zagonetke' se nalazi na stranici br. 167.*

|  |  |
| --- | --- |
| Rješenje zadatka br. **130**: | |
| 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14: | void main()  {  int a = 23;  int b = 24;  int temp;  temp = a;  a = b;  b = temp;  cout << "a: " << **hex** << a << ", " << **oct** << a << ", " << **dec** << a << endl;  cout << "b: " << **hex** << b << ", " << **oct** << b << ", " << **dec** << b << endl;  } |

|  |  |
| --- | --- |
| Rješenje zadatka br. **131**: | |
| 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12: | void main()  {  int a,b,c;    a = **0x**123; //hexadecimalno '123'  b = **0**123; //oktalno '123'  c = 123; //decimalno '123'  cout << "a: " << a << endl;  cout << "b: " << b << endl;  cout << "c: " << c << endl  } |

|  |
| --- |
| Rješenje zadatka br. **132:** *(u skladu sa fazama programiranja)*: |
| **1. Analiza i specifikacija:**   |  |  | | --- | --- | | **Input** | **Output** | | Broj za početak - parametar (m) | Razlomak (z) | | Broj za kraj parametar (n) | | Nazivnik (x) |   Ograničenja: nazivnik **x mora biti različit od nule**  **2. Dizajn**   |  |  |  |  | | --- | --- | --- | --- | | **Podatkovni objekt** | **Vrsta vrijednosti** | **Tip objekta** | **Ime objekta** | | Broj za parametar | varijabla | int | m | | Broj za parametar | varijabla | int | n | | Nazivnik | varijabla | int | x | | Razlomak | varijabla | **float** | z | | Suma kubova | varijabla | int | suma |     **3. Algoritam – blok dijagram**   |  | | --- | |  |   **4. Kôd (code)**   |  |  | | --- | --- | | 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24:  25:  26:  27:  28:  29:  30:  31:  32:  33:  34:  35:  36:  37:  38:  39:  40:  41:  42:  43:  44: | #include <iostream>  using namespace std;  int ZbirKubova(int,int); //deklaracija (prototip) funkcije  void main()  {  int m,n,x;  float z;  char pitanje;  do  {  cout << "Unesite m, m \n";  cin >> m >> n;  do  {  cout << "Unesite x (x mora biti razlicito od 0) \n";  cin >> x;  }while(x==0); // ponavlja sve dok je x == 0, tj. tek kad  // bude x != 0 napusti petlju  **z** = ZbirKubova(m,n)/ float(x\*x); // pazite: int/int daje int: cout << "z = " << **z** << endl; // procitajte str. 12 (Workshop2)    cout << "Ponoviti? (D)a...\n";  cin >> pitanje;  }while (pitanje=='d' || pitanje=='D'); // ponovlja čitav program ako je korisnik unio 'd'  } // kraj funkcije main – kraj programa  int ZbirKubova(int u1, int u2) //definicija funkcije  {  int suma;  suma = 0;  for (int i=u1; i<=u2; i++)  {  suma = suma + i\*i\*i;  }  return suma; //izlaz iz funkcije je suma kubova  } | |

|  |  |
| --- | --- |
| Rješenje zadatka br. **133**: | |
| 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24:  25:  26:  27:  28:  29:  30:  31:  32:  33:  34:  35:  36:  37:  38:  39:  40:  41:  42:  43:  44:  45: | #include <iostream>  using namespace std;  void ZbirKubova(int, int, int**&** );  void main()  {  int m, n, x, zbir;  float z;  char pitanje;  do  {  cout << "Unesite m, m \n";  cin >> m >> n;  do  {  cout << "Unesite x (x mora biti razlicito od 0) \n";  cin >> x;  }while(x==0);    ZbirKubova(m, n, **zbir**); //funkcija je tipa void  z = **zbir** / float(x\*x);  cout << "z = " << z << endl;    cout << "Ponoviti? (D)a...\n";  cin >> pitanje;  }while (pitanje=='d' || pitanje=='D');  } // kraj funkcije main – kraj programa  void ZbirKubova(int u1, int u2, int**&** Izlaz)  {  int suma;  suma = 0;  for (int i=u1; i<=u2; i++)  {  suma = suma + i\*i\*i;  }  **Izlaz** = suma; //izlaz iz funkcije je suma kubova  } |

|  |  |
| --- | --- |
| Rješenje zadatka br. **134**: | |
| 1:  2:  3:  4:  5:  6:  7:  8:  9:  10:  11:  12:  13:  14:  15:  16:  17:  18:  19:  20:  21:  22:  23:  24:  25:  26:  27:  28:  29:  30:  31:  32:  33:  34:  35:  36:  37:  38:  39:  40:  41:  42:  43:  44:  45:  46:  47:  48:  49:  50:  51:  52:  53:  54:  55:  56:  57:  58:  59: | #include <iostream>  using namespace std;  const float Pi = 3.141; // konstantna globalna varijabla  void f1(float a)  {  cout << " Povrsina kvadrata = " << a\*a << endl;  }  void f2(float a, float b)  {  cout << " Povrsina pravougaonika = " << a\*b << endl;  }  void f3(float r)  {  cout << " Povrsina kruga = " << r\*r\*Pi << endl;  }  void main()  {  int izbor;  float a, b;  do  {  cout << "\n\n=========\n";  cout << "IZBORNIK \n";  cout << "1: kvadrat \n";  cout << "2: pravougaonik \n";  cout << "3: krug \n";  cout << "4: izlaz \n>>>";  cin >> izbor;  switch(izbor)  {  case 1:  cout << "Unesite a: ";  cin >> a;  f1(a);  break;  case 2:  cout << "Unesite a i b \n";  cin >> a >> b;  f2(a, b);  break;  case 3:  cout << "Unesite r: ";  cin >> a;  f3(a);  break;  case 4:  break;  default:  cout << "Pogresan izbor \n";  }; // kraj switch-iskaza  }while(izbor != 4);  } |