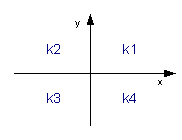
# Zadatak 1a:

Deklarišite strukturu kružnica i instancirajte objekte k1, k2, k3 i k4. Dodijelite vrijednosti obilježjima tako:  
a) da se centar kružnice k1 nalaziti u prvom kvadrantu koordinatnog sistema, k2 u drugom, k3 u trećem, a k4 u četvrtom kvadrantu (na desnoj slici su označena sva četiri kvadranta)  
b) sa tastature učitajte vrijednost poluprečnika kružnice k1   
c) neka poluprečnik kružnice k2 bude iste veličine kao kod k1  
d) neka poluprečnik kružnice k3 bude duplo veći od k2  
e) neka poluprečnik kružnice k4 bude zbir poluprečnika od k2 i k3



A zatim, ispišite karakteristike kružnica (obilježja objekata k1, …,k4) koristeći pune rečenice, kao u sljedećem primjeru:

- Kružnica iz drugog kvadranta sa centrom O(3,4) ima isti poluprečnik kao kružnica O(-4,6) iz prvog kvadranta koji iznosi r = 2.

- Kružnica O(─9,─20) iz trećeg kvadranta ima duplo veći   
- poluprečnik (r = 4) od kružnice O(-4,6).

- Kružnica O(11,-23) ima poluprečnik (r = 6) koji predstavlja zbir poluprečnika kružnica O(-9,-20) i O(11,-23)

|  |
| --- |
| **Rješenje** |
| #include <iostream>  using namespace std;  struct kruznica  {  float x;  float y;  float poluprecik;  };  void main()  {  kruznica k1, k2, k3, k4;  k1.x = 3;  k1.y = 4;  cout << "Unesi r1: ";  cin >> k1.poluprecik;  k2.x = -4;  k2.y = 6;  k2.poluprecik = k1.poluprecik;  k3.x = -9;  k3.y = -20;  k3.poluprecik = k1.poluprecik \* 2;  k4.x = 11;  k4.y = -23;  k4.poluprecik = k1.poluprecik + k2.poluprecik;  cout << "Kružnica iz drugog kvadranta sa centrom O(";  cout << k1.x << "," << k1.y << ") ima isti poluprečnik kao \nkružnica O(";  cout << k2.x << "," << k2.y << ") iz prvog kvadranta koji iznosi r = ";  cout << k1.poluprecik;  //....  } |

# Zadatak 1b:

Prepravite prethodni zadatak tako da koristite strukturu Tacka sa atributima int x, int y

|  |
| --- |
| **Rješenje** |
| #include <iostream>  using namespace std;  struct Tacka  {  float x;  float y;  };  struct Kruznica  {  Tacka centar;  float poluprecnik;  };  void main()  {  Kruznica k1;  Kruznica k2;  Kruznica k3;  Kruznica k4;  //a  k1.centar.x = 5;  k1.centar.y = 6;  k2.centar.x = -5;  k2.centar.y = 6;  k3.centar.x = -5;  k3.centar.y = -6;  k4.centar.x = 5;  k4.centar.y = -6;  //b  cout << "Unesite poluprecnik za k1: ";  cin >> k1.poluprecnik;  //c  k2.poluprecnik = k1.poluprecnik;  //d  k3.poluprecnik = 2 \* k2.poluprecnik;  //e  k4.poluprecnik = k2.poluprecnik + k3.poluprecnik;  cout << endl;  } |

# Zadatak 2:

Deklarišite strukturu trougao i kruznica:

a. kreirajte troguao T1 čije ćete koordinate tačaka A, B i C učitati sa tastature   
b. kreirajte kružnicu K1, čiji se centar nalazi u tački A trougla T1   
c. vrijednost poluprečnika kružnice K1 učitajte sa tastature  
d. kreirajte kružnicu K3 sa istim osobinama kao kružnica K1, tj. K1 iskopirajte u novi K3  
e. povećajte poluprečnik kružnice K3 za 10%  
f. kreirajte kružnicu K2 koja se nalazi na tački B sa istim poluprečnikom kao kod K3

|  |
| --- |
| **Rješenje** |
| #include <iostream>  using namespace std;  struct kruznica  {  float x;  float y;  float poluprecik;  };  struct trougao  {  float a\_x;  float a\_y;  float b\_x;  float b\_y;  float c\_x;  float c\_y;  };  void main()  {  trougao T1;  cout << "unesi x1 i y1 za tacku A \n";  cin >> T1.a\_x >> T1.a\_y;  cout << "unesi x2 i y2 za tacku B \n";  cin >> T1.b\_x >> T1.b\_y;  cout << "unesi x3 i y3 za tacku C \n";  cin >> T1.c\_x >> T1.c\_y;  kruznica K1;  K1.x = T1.a\_x;  K1.y = T1.a\_y;  cout << "Unesite r za K1 \n";  cin >> K1.poluprecik;  kruznica K3;  // kopiranje cijelog objekta:  K3 = K1;  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Kopiranje svih obiljezja iz K1 u K3 (umjesto linije br. 46)  K3.x = K1.x;  K3.y = K1.y;  K3.poluprecik = K1.poluprecik;  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  K3.poluprecik = K3.poluprecik \* 1.1;  kruznica K2;  K2.x = T1.b\_x;  K2.y = T1.b\_y;  K2.poluprecik = K3.poluprecik;  } |

# Zadatak 3:

Nastavlja se Zadatak 2

Napravite funkciju void infoKruznica, čiji je ulazni parametar podatak tipa kruznica. Funkcija treba ispisati osobine kružnice u sljedećem obliku:

infoKruznica  
centar O(1.4, 2.5)   
poluprečnik r = 12

Zatim, iskoristite ovu funkciju da bi ispisali osobine kružnice K1 i K2 iz zadatka br. 2.

|  |
| --- |
| **Rješenje** |
| #include <iostream>  using namespace std;  struct Tacka  {  float x;  float y;  };  struct Kruznica  {  Tacka centar;  float poluprecnik;  };  struct Trougao  {  Tacka A;  Tacka B;  Tacka C;  };  void info(Kruznica k)  {  cout << "centar O( " << k.centar.x << ", " << k.centar.y << ")";  cout << "poluprečnik r = " << k.poluprecnik << endl;  }  void main()  {  //a  Trougao T1;  cout << "Unesite x i y za tacku A: ";  cin >> T1.A.x;  cin >> T1.A.y;  cout << "Unesite x i y za tacku B: ";  cin >> T1.B.x;  cin >> T1.B.y;  cout << "Unesite x i y za tacku C: ";  cin >> T1.C.x;  cin >> T1.C.y;  //b  Kruznica K1;  K1.centar = T1.A;  //ili  K1.centar.x = T1.A.x;  K1.centar.y = T1.A.y;  //c  cout << "Unesite poluprecnik za K1: ";  cin >> K1.poluprecnik;  //d  Kruznica K3 = K1;  //e  K3.poluprecnik = K3.poluprecnik \* 1.1;  //ili  K3.poluprecnik \*= 1.1;  //f  Kruznica K2 = { T1.B, K3.poluprecnik };  //ili  K2.centar = T1.B;  K2.poluprecnik = K3.poluprecnik;  info(K1);  info(K2);  info(K3);  } |

Strukture odvojite u zasebni header fajlove.

# Zadatak 4a - Datum

1. Deklarisati strukturu **Datum** sa atributima d, m i g.
2. Implementirati funkciju koja ispisuje podatke o datumu **date**

void info(Datum date)

1. Implementirati funkciju koja vraća objekat tipa Datum sa vrijednostima učitanih sa tastature

Datum ucitaj\_datum()

|  |
| --- |
| **Rješenje** |
| // stuktura Datum  struct Datum {  int d;  int m;  int g;  };  void info(Datum date) {  // dd/mm/gg  cout << date.d << "/" << date.m << "/" << date.g << endl;  }  Datum ucitaj\_datum()  {  Datum x;  cout << "unesite dan, mjesec i godinu: ";  cin >> x.d;  cin >> x.m;  cin >> x.g;  return x;  } |

# Zadatak 4b – Osoba

1. Deklarisati strukturu **Osoba** sa atributima id, ime.
2. Implementirati funkciju koja ispisuje podatke o osobi **o**
3. void info(Osoba o)

|  |
| --- |
| **Rješenje 1 i 2** |
| // struktura Osoba  struct Osoba {  int id;  char ime[15];  };  void info(Osoba o) {  cout << "Ime osobe je: " << o.ime << " id= " << o.id << endl;  } |

1. Alocirajte 5 objekatat (kao globalne varijable) tipa osoba (o1, o2, o3, o4, o5) i dodjelite neke vrijednosti.
2. Alocirajte niz tipa osoba veličine 5.

const int max\_s = 5;

Osoba ispitivaci [max\_s] = {o1, o2, o3, o4, o5}

|  |
| --- |
| **Rješenje c i d – sa varijablama o1, ... o5** |
| Osoba o1 = { 534, "Nina B." };  Osoba o2 = { 435, "Jasmin A."};  Osoba o3 = { 256, "Denis M." };  Osoba o4 = { 325, "Emina J." };  Osoba o5 = { 914, "Zanin V." };  const int max\_i = 5;  Osoba ispitivaci[max\_i] = { o1, o2, o3, o4, o5,}; |
| **Rješenje c i d – bez varijabli o1, ... o5** |
| const int max\_i = 5;  Osoba ispitivaci[max\_i] = {  { 534, "Nina B." },  { 435, "Jasmin A." },  { 256, "Denis M." },  { 325, "Emina J." },  { 914, "Zanin V." },  }; |

1. Implementirati funkciju koja vraća objekat iz niza **studenti** na osnovu rednog broja kojeg korisnik unese pomoću tastature.

|  |
| --- |
| Spisak osoba:  1. Osoba A B, ID=302  2. Osoba C B, ID=303  3. Osoba D B, ID=304  4. Osoba E B, ID=305  5. Osoba F B, ID=306  Unesite RB. |

Osoba odaberite\_osobu()

|  |
| --- |
| **Rješenje e** |
| // Studenti  const int max\_s = 5;  Osoba studenti[max\_s] = {  { 1, "Marko" },  { 2, "Darko" },  { 3, "Zarko" },  { 4, "Adil" },  { 5, "Faruk" }  };  Osoba odaberite\_osobu()  {  int a;  cout << "Spisak osoba: " << endl;  for (int i = 0; i < max\_i; i++)  {  cout << i << "Ime osobe: " << ispitivaci[i].ime << ", ID: " <<  ispitivaci[i].id << endl;  }  cout << "Unesite RB: ";  cin >> a;  Osoba x = ispitivaci[a];  return x;  } |

1. U funkciji main implementirajte sljedeće

* Deklarište datume **d1**, **d2 i d3** i učitajte vrijednosti pomoću funkcije ucitaj\_datum
* Ispišite datume **d1, d2 i d3** pomoću funkcije info(datum)
* Odaberite studenta s1 i ispišite podatke pomoću funkcije info(osoba)

|  |
| --- |
| **Rješenje f** |
| Datum d1 = ucitaj\_datum();  Datum d2 = ucitaj\_datum();  Datum d3 = ucitaj\_datum();  info(d1);  info(d2);  info(d3);  Osoba s1 = odaberite\_osobu(studenti, max\_s);  info(s1); |

Prepravite prethodno rješenje

1. Prepravite funkciju odaberite\_osobu tako funkcija koristi različite nizove tipa osoba (npr. ispitivače i studente)

Osoba odaberite\_osobu() -> Osoba odaberite\_osobu(Osobe osobe[], int v)

|  |
| --- |
| **Rješenje g** |
| Osoba odaberite\_osobu(Osoba niz[], int z)  {  int a;  cout << "Spisak osoba: " << endl;  for (int i = 0; i < z; i++)  {  cout << i << "Ime osobe: " << niz[i].ime << ", ID: " <<  niz[i].id << endl;  }  cout << "Unesite RB: ";  cin >> a;  Osoba x = niz[a];  return x;  } |

# Zadatak 4c - Ispit

Implementirajte strukturu Ispit

|  |
| --- |
| struct Ispit  {  Osoba ispitivac;  Datum datum\_ispita;  char predmet[10];  };  Ispit ucitaj\_ispit();  void prikazi\_ispite(Ispit ispiti[], int max); |

|  |
| --- |
| **Rješenje** |
| struct Ispit {  Osoba ispitivac;  Datum datum\_ispita;  char predmet[10];  };  Ispit ucitaj\_ispit()  {  Ispit x;  cout << "Unesite predmet " << endl;  cin >> x.predmet;  cout << "Odaberite ispitivaca" << endl;  x.ispitivac = odaberite\_osobu(ispitivaci, max\_i);  cout << "Unesite datum ispita" << endl;  x.datum\_ispita = ucitaj\_datum();  return x;  }  void prikazi\_ispite(Ispit ispiti[], int max)  {  for (int i = 0; i < max; i++)  {  cout << i +1 << ". ";  info(ispiti[i].datum\_ispita);  info(ispiti[i].ispitivac);  cout << "Predmet -> " << ispiti[i].predmet << endl;  }  }  Ispit& odaberite\_ispit(Ispit ispiti[], int max)  {  prikazi\_ispite(ispiti, max);  cout << "Unesite RB ispita: ";  int x;  cin >> x;  return ispiti[x - 1];  } |

Program za testiranje

|  |
| --- |
| void main()  {  int \_brojac\_ispita = 0;  const int max\_ispita = 30;    Ispit ispiti[max\_ispita];  ispiti[\_brojac\_ispita++] = ucitaj\_ispit();  ispiti[\_brojac\_ispita++] = ucitaj\_ispit();  ispiti[\_brojac\_ispita++] = ucitaj\_ispit();  ispiti[\_brojac\_ispita++] = ucitaj\_ispit();  Ispit& i = odaberite\_ispit(ispiti, \_brojac\_ispita); //korisnik bira ispit  i.datum\_ispita.d++; //pomjeri datum ispita na naredni dan  prikazi\_ispite(ispiti, \_brojac\_ispita);  } |

# Zadatak 4d – Prijava za ispit

Dodajte strukturu **PrijavaZaIspit**.

U strukturu Ispit dodajte niz tipa **PrijavaZaIspit**

|  |
| --- |
| struct PrijavaZaIspit  {  Osoba student;  Datum datum\_prijave;  Datum datum\_odjave;  int ocjena;  };  const int max\_p = 100;  struct Ispit  {  Osoba ispitivac;  Datum datum\_ispita;  char predmet[10];  int brojac\_prijavljenih;  PrijavaZaIspit prijave[max\_p];  };  void info(PrijavaZaIspit p)  {  cout << "Student: ";  info(p.student);  cout << " Datum prijave: ";  info(p.datum\_prijave);  cout << " Datum odjave: ";  info(p.datum\_odjave);  cout << endl;  } |

Prilagodite funkcijevoid ucitaj\_ispit() i void prikazi\_ispite(Ispit[], int) shodno novim modifikacijima u strukturi Ispit.

|  |
| --- |
| Rješenje |
| Ispit ucitaj\_ispit()  {  Ispit x;  cout << "Unesite predmet " << endl;  cin >> x.predmet;  cout << "Odaberite ispitivaca" << endl;  x.ispitivac = odaberite\_osobu(ispitivaci, max\_i);  cout << "Unesite datum ispita" << endl;  x.datum\_ispita = ucitaj\_datum();    x.brojac\_prijavljenih = 0;  return x;  }  void prikazi\_ispite(Ispit ispiti[], int max)  {  for (int i = 0; i < max; i++)  {  info(ispiti[i].datum\_ispita);  info(ispiti[i].ispitivac);  cout << "Predmet -> " << ispiti[i].predmet ;  cout << ", \tBroj prijavljenih: " << ispiti[i].brojac\_prijavljenih;  }  } |

# Zadatak 4e – Glavni program

Dovršite program

|  |
| --- |
| void main()  {  const int max\_ispiti = 10;  Ispit ispiti[max\_ispiti];  int brojac\_ispita = 0;  int x;  do  {  cout << "1. Ispiti - prikazi sve" << endl;  cout << "2. Ispiti - dodaj novi (zakazi novi ispit)" << endl;  cout << "3. Prijavljeni za ispit: prikazi sve (za ispit x)" << endl;  cout << "4. Prijavljeni za ispit: dodaj novu prijavu (za ispit x)" << endl;  cout << "5. Evidentiraj sve ocjene (za ispit x)" << endl;  cout << "0. EXIT" << endl;  cin >> x;  if (x == 1)  {  }  if (x == 2)  {    if (x == 3)  {  }  if (x == 4)  {  }  if (x == 5)  {  }  } while (x != 0);  } |

**Rješenje**

|  |
| --- |
| #include <iostream>  using namespace std;  // stuktura Datum  struct Datum {  int d;  int m;  int g;  };  void info(Datum date) {  // dd/mm/gg  cout << date.d << "/" << date.m << "/" << date.g << ". ";  }  Datum ucitaj\_datum()  {  Datum x;  cout << "unesite dan, mjesec i godinu: ";  cin >> x.d;  cin >> x.m;  cin >> x.g;  return x;  }  // struktura Osoba  struct Osoba {  int id;  char ime[15];  };  void info(Osoba o) {  cout << "Ime osobe je: " << o.ime << " id= " << o.id << ". ";  }  // Studenti  const int max\_s = 5;  Osoba studenti[max\_s] = {  { 1, "Marko" },  { 2, "Darko" },  { 3, "Zarko" },  { 4, "Adil" },  { 5, "Faruk" }  };  Osoba odaberite\_osobu(Osoba niz[], int z)  {  int a;  cout << "Spisak osoba: " << endl;  for (int i = 0; i < z; i++)  {  cout << i << "Ime osobe: " << niz[i].ime << ", ID: " << niz[i].id << endl;  }  cout << "Unesite RB: ";  cin >> a;  Osoba x = niz[a];  return x;  }  Osoba o1 = { 534, "Nina B." };  Osoba o2 = { 435, "Jasmin A."};  Osoba o3 = { 256, "Denis M." };  Osoba o4 = { 325, "Emina J." };  Osoba o5 = { 914, "Zanin V." };  const int max\_i = 5;  Osoba ispitivaci[max\_i] = { o1, o2, o3, o4, o5,};  struct PrijavaZaIspit  {  Osoba student;  Datum datum\_prijave;  Datum datum\_odjave;  int ocjena;  };  const int max\_p = 100;  struct Ispit {  Osoba ispitivac;  Datum datum\_ispita;  char predmet[10];  int brojac\_prijavljenih;  PrijavaZaIspit prijave[max\_p];  };  Ispit ucitaj\_ispit()  {  Ispit x;  cout << "Unesite predmet " << endl;  cin >> x.predmet;  cout << "Odaberite ispitivaca" << endl;  x.ispitivac = odaberite\_osobu(ispitivaci, max\_i);  cout << "Unesite datum ispita" << endl;  x.datum\_ispita = ucitaj\_datum();    x.brojac\_prijavljenih = 0;  return x;  }  void info(PrijavaZaIspit p)  {  cout << "Student: ";  info(p.student);  cout << " Datum prijave: ";  info(p.datum\_prijave);  cout << " Datum odjave: ";  info(p.datum\_odjave);  cout << endl;  }  void prikazi\_ispite(Ispit ispiti[], int max)  {  for (int i = 0; i < max; i++)  {  cout << i +1 << ". ";  info(ispiti[i].datum\_ispita);  info(ispiti[i].ispitivac);  cout << "Predmet -> " << ispiti[i].predmet ;  cout << ", \tBroj prijavljenih: " << ispiti[i].brojac\_prijavljenih << endl;  }  }  Ispit& odaberite\_ispit(Ispit ispiti[], int max)  {  prikazi\_ispite(ispiti, max);  cout << "Unesite RB ispita: ";  int x;  cin >> x;  return ispiti[x - 1];  }  void main()  {  const int max\_ispiti = 10;  Ispit ispiti[max\_ispiti];  int brojac\_ispita = 0;  int x;  do  {  cout << "1. Ispiti - prikazi sve" << endl;  cout << "2. Ispiti - dodaj novi (zakazi novi ispit)" << endl;  cout << "3. Prijavljeni za ispit: prikazi sve (za ispit x)" << endl;  cout << "4. Prijavljeni za ispit: dodaj novu prijavu (za ispit x)" << endl;  cout << "5. Evidentiraj sve ocjene (za ispit x)" << endl;  cout << "0. EXIT" << endl;  cin >> x;  if (x == 1)  {  prikazi\_ispite(ispiti, brojac\_ispita);  }  if (x == 2)  {  Ispit i = ucitaj\_ispit();  ispiti[brojac\_ispita] = i;  brojac\_ispita++;  }  if (x == 3)  {  Ispit i = odaberite\_ispit(ispiti, brojac\_ispita);  for (int j = 0; j<i.brojac\_prijavljenih; j++)  {  PrijavaZaIspit& p = i.prijave[j];  info(p);  }  }  if (x == 4)  {  Ispit& i = odaberite\_ispit(ispiti, brojac\_ispita);  PrijavaZaIspit novaP;  novaP.student = odaberite\_osobu(studenti, max\_s);  novaP.datum\_prijave = ucitaj\_datum();  i.prijave[i.brojac\_prijavljenih++] = novaP;  }  if (x == 5)  {  Ispit& i = odaberite\_ispit(ispiti, brojac\_ispita);  for (int j = 0; j<i.brojac\_prijavljenih; j++)  {  info(i.prijave[j].student);  cout << ", Unesite ocjenu sa ispita: ";  cin >> i.prijave[j].ocjena;  }  }  } while (x != 0);  } |

Autor:

mr. Adil Joldić

[adil@edu.fit.ba](mailto:adil@edu.fit.ba)