

Priprema za sedme vježbe iz Programiranja

Jednostavno pretraživanje i sortiranje polja

Napomena: proučiti materijale vezane uz sedmo predavanje

Napišite program u C++ koji će korisniku staviti na raspolaganje izbor sa sljedećim mogućnostima:

1. Pretraživanje polja.
2. Sortiranje polja brojeva metodom Bubble sort
3. Sortiranje polja brojeva metodom izbora (Selection sort)
4. Sortiranje polja prezimena i imena metodom umetanja (Insertion sort)
9. Izlaz iz programa

Potrebno je korisniku ispisati mogućnosti, te omogućiti unos izbora. Izbor mogućnosti treba implementirati pomoću selekcije tipa **switch** (svaki pojedini **case** odnosi se na pojedinu mogućnost). Korisnik može više puta birati mogućnost, tako dugo dok ne izabere izlaz iz programa tj. selekcija tipa **switch** treba se nalaziti unutar iteracije tipa **do-while**. Potrebno je brojati izbore korisnika (koristiti polje brojača).

Za 1 bod:

Potrebno je formirati iteraciju tipa **do-while** unutar koje se korisniku ispisuju mogućnosti i selekciju tipa **switch**, te prepoznati neispravan unos (= ako korisnik unese izbor koji nije pokriven odgovarajućim **case**-om, ispisati poruku da je izbor neispravan). Pri izlazu iz programa potrebno je ispisati ukupan broj poziva svake od mogućnosti programa (unutar raspona 1-4, odnosno 9), te broj neispravnih izbora (koristiti polje s brojačima). Nakon toga treba ispisati koji je od izbora bio najčešći (odnosno više njih, ako se pojavljuju jednaki broj puta).

Za 2 boda:

Mogućnost 1. Definirajte polje od 1000 elemenata tipa **int**. Program treba upitati korisnika za broj elemenata polja (**N**) koji će se generirati (mora biti u rasponu 1-1000; ako nije, zatražiti ponovni unos od korisnika) i za broj koji se traži u polju (**A**). Nakon toga program pomoću generatora slučajnih brojeva generira **N** elemenata polja, svaki u rasponu 1-100 i ispisuje ih na ekran. Na kraju program ispisuje poruku o tome koliko puta se broj **A** pojavljuje u polju, odnosno, da se ne pojavljuje u polju.

Za 3 boda:

Mogućnost 2. Definirajte polje od 1000 elemenata tipa **float**. Program treba upitati korisnika za broj elemenata polja (**N**) koji će se generirati (mora biti u rasponu 1-1000; ako nije, zatražiti ponovni unos od korisnika). Potrebno je generirati **N** elemenata polja, svaki u rasponu 50-100, na dvije decimale i ispisati ih na ekran. Nakon toga polje treba sortirati uzlazno metodom **Bubble sort** i ispisati ga sortiranog. Na kraju treba ispisati broj usporedbi elemenata polja i broj njihovih zamjena u postupku sortiranja.

Za 4 boda:

Mogućnost 3. Definirajte polje od 100 elemenata tipa *char*. Program treba upitati korisnika za broj elemenata polja (*N*) koji će se generirati (mora biti u rasponu 1-100; ako nije, zatražiti ponovni unos od korisnika). Potrebno je pomoću generatora slučajnih brojeva generirati *N* elemenata polja, svaki u rasponu slova engleske abecede (*A-Z*), i ispisati ih na ekran. Nakon toga polje treba sortirati uzlazno po abecedi metodom izbora (*Selection sort*) i ispisati ga sortirano. Na kraju treba ispisati broj usporedbi elemenata polja i broj njihovih zamjena u postupku sortiranja.

Za 5 bodova:

Mogućnost 4. Definirajte polje za smještanje najviše 100 znakovnih nizova (svaki dužine najviše 40 znakova), koje će sadržavati prezimena i imena. Treba upitati korisnika za broj prezimena i imena koliko namjerava unijeti (*N*; mora biti u rasponu 1-100; ako nije, zatražiti ponovni unos od korisnika). Nakon toga, korisnik treba unijeti *N* prezimena i imena u polje*. Polje zatim treba sortirati uzlazno po abecedi metodom umetanja (*Insertion sort*)**. Na kraju treba sortirano polje ispisati.

* Koristiti *cin.getline* za unos znakovnih nizova

** Za usporedbu znakovnih nizova po abecedi koristiti funkciju *strcmp* iz biblioteke *cstring*