

Priprema za petu vježbu iz kolegija Programiranje

Iteracije while i do while te složeni logički izrazi

Priprema za vježbu: proučiti materijale vezane uz peto predavanje

Napomena: priprema za petu vježbu donosi najviše 5 bodova dok peta vježba donosi najviše 3 boda

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

1. Pogađanje broja
2. Pogađanje koordinata
3. Površina trokuta
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). Također, 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 ispravne, odnosno neispravne izbore korisnika.

Za 1 bod:

- ispisati mogućnosti, formirati iteraciju tipa *do-while* i selekciju tipa *switch*, te prepoznati neispravan unos (= ako korisnik unese izbor koji nije pokriven odgovarajućim *case*-om, ispisati poruku da je izbor neispravan). **Napomena:** pojedini *case*-ovi mogu za početak biti prazni. Pri izlazu iz programa potrebno je ispisati ukupan broj ispravnih izbora korisnika (1-3, odnosno 9), te ukupan broj neispravnih izbora (izvan raspona 1-3, odnosno 9).

Za 2 boda:

Mogućnost 1. Uz pomoć vremenske funkcije postaviti sjeme generatora slučajnih brojeva. Uz pomoć generatora slučajnih brojeva računalno treba izabrati slučajan broj (A) u rasponu 0-100. Korisnik zatim pogađa broj. Ako pogodi, ispiše se poruka da je broj pogođen u prvom pokušaju i slijedi povratak u glavni izbornik. U suprotnom, program ulazi u iteraciju tipa *while* gdje treba sljedeće:

- ispisati korisniku poruku o tome da li je uneseni broj veći ili manji od traženog (A),
- zatražiti od korisnika da ponovo pokuša pogoditi broj,
- u slučaju pogotka ispisati poruku da je broj pogođen, te ukupan broj pokušaja. Nakon toga treba se vratiti u glavni izbornik,
- u slučaju da je broj pokušaja veći od 10, ispisati poruku da broj nije pogođen i vratiti se u glavni izbornik.

Za 3 boda:

Mogućnost 2. Uz pomoć vremenske funkcije postaviti sjeme generatora slučajnih brojeva. Uz pomoć generatora slučajnih brojeva izabrati X i Y koordinatu točke u koordinatnom sustavu,

tako da su obje koordinate u rasponu 0-1000. Potrebno je formirati iteraciju tipa **for** s 10 ponavljanja unutar koje korisnik pokušava što preciznije pogoditi koordinate točke:

- korisnik unosi koordinate X1 i Y1, unutar iteracije tipa **do-while**, tako dugo dok ne unese ispravne vrijednosti (unutar raspona 0-1000)
- udaljenost točke (D) od one koju je izabralo računalo računa se prema Pitagorinom poučku:

$$D = \sqrt{(X1 - X) * (X1 - X) + (Y1 - Y) * (Y1 - Y)}$$

- korisniku ispisati udaljenost točaka (tj. veličinu promašaja).
- ako je $D < 50$ ispisati da je točka pogođena, broj pokušaja, te se vratiti u glavni izbornik
- ako je broj pokušaja veći od 10 tada ispisati poruku da točka nije pogođena i vratiti se u glavni izbornik

Za 4 boda:

Mogućnost 2 proširiti, tako da se korisniku nakon potrošenih 10 pokušaja ispiše podatak o tome kolika je bila prosječna udaljenost do tražene točke (aritmetička sredina udaljenosti).

Za 5 bodova:

Mogućnost 3. Površinu trokuta možemo izračunati prema Heronovoj formuli (a, b, c su duljine pojedinih stranica trokuta):

$$s = \frac{a + b + c}{2}$$

$$P = \sqrt{s * (s - a) * (s - b) * (s - c)}$$

Ukoliko je vrijednost pod korijenom manja ili jednaka nuli, tada stranice ne tvore trokut!

Formirajte vanjsku iteraciju tipa **do-while**, unutar koje će korisnik unositi duljine stranica, tako dugo dok ne unese nulu za sve duljine ($a=0$, $b=0$ i $c=0$). Unutar te petlje formirajte unutarnju iteraciju tipa **do-while**, unutar koje program ostaje tako dugo dok stranice ne čine trokut (a nisu ni unesene nule za duljine stranica!):

- korisniku se ispiše redni broj unosa (broje se samo unosi gdje stranice čine trokut)
- korisnik unosi duljine stranica **a, b, c** (realni podaci tipa **float**)
- ako je neka od stranica manja ili jednaka nuli, ispisati poruku i zatražiti ponovni upis
- ako stranice ne čine trokut, ispisati odgovarajuću poruku korisniku, te da ponovi unos

Po izlasku iz unutarnje iteracije tipa **do-while** izračunati i ispisati površinu trokuta.

Po izlasku iz vanjske iteracije ispisati površinu najvećeg unesenog trokuta, te redni broj unosa za taj trokut.