

## Laboratorijske vježbe 6.

- 5.1 Na ekranu iscrtati šahovsku tablu na kojoj se nalaze dvije kraljice: prva na poziciji  $(m, n)$ ; druga na poziciji  $(p, q)$ . Zadane položaje kraljica označiti sa „k“, polja na koja može ići kraljica sa pozicije  $(m, n)$  a koja nisu napadnuta kraljicom na poziciji  $(p, q)$  označiti sa „\*“ a sva ostala polja sa „0“. Na primjer za  $(m, n)=(1, 1)$  i  $(p, q)=(7, 8)$  šahovska tabla treba da izgleda ovako

```
k 0 * * * * 0
* * 0 0 0 0 0
* 0 * 0 0 0 0
* 0 0 * 0 0 0
* 0 0 0 * 0 0
* 0 0 0 0 * 0 0
0 0 0 0 0 0 k
* 0 0 0 0 0 0
```

- 5.2 Napisati program koji traži da se unese prirodan broj  $n$  a zatim niz od  $n$  prirodnih brojeva, koji treba da modifikuje dati niz na način da svaki element niza treba zamjeniti sa prvim sljedećim elementom koji je veći od njega. Ako ne postoji takav element poslije njega taj broj treba zamjeniti sa nulom. Na kraju treba ispisati niz nakon modifikacije. Na primjer, za ulaz  $n=7$  i niz 2, 8, 3, 7, 5, 4, 6 izlaz treba biti niz 8, 0, 7, 0, 6, 6, 0, a za ulaz  $n=7$  i niz 7, 6, 5, 4, 6, 5, 9 izlaz treba biti niz 9, 9, 6, 6, 9, 9, 0.

- 5.3 Napisati funkciju “Tougao” sa tri parametra “Visina“, “Znak1” i “Znak2” koja iscrtava na ekranu jednakokraki trougao sa osnovicom okrenutom nadolje čija unutrašnjost *nije ispunjena*. Visina trougla (u znakovima) određena je prvim parametrom, znak koji se koristi za iscrtavanje određen je drugim parametrom, a visina se iscrtava znakom koji je određen trećim parametrom. Na primjer, ako funkciju pozovemo sa

```
Tougao (6, '#', '+');
```

na ekranu treba da dobijemo sljedeći prikaz:

```
      #
     ##
    # + #
   # + #
  # + #
#####
```

Predvidjeti i mogućnost da se drugi parametar može izostaviti. U tom slučaju kao znak za iscrtavanje treba koristiti zvjezdicu. Obavezno napisati i kratki testni program (“main” funkciju) u kojem ćete demonstrirati napisanu funkciju.

- 5.4 Još je Heronu prije 2000 godina bio poznat sljedeći postupak (algoritam) za računanje kvadratnog korijena proizvoljnog broja  $x$ : formira se niz brojeva  $a_0, a_1, a_2$  itd. po sljedećem pravilu:

$$a_0 = 1; \quad a_{k+1} = (a_k + x/a_k)/2 \quad \text{za } k > 0$$

Ovaj niz konvergira vrlo brzo ka korijenu iz  $x$ . U praksi je dovoljno izračunati samo nekoliko elemenata ovog niza, jer se vrlo brzo elementi počinju praktično ponavljati (sa onolikom tačnošću koliko dopušta realni tip podataka). Tada postupak možemo obustaviti, i posljednju izračunatu vrijednost  $a_k$  proglasiti traženim korijenom. Napraviti funkciju “korijen” koja računa korijen svog argumenta Heronovim postupkom. Za realizaciju funkcije *ne koristiti nizove*.

Napisanu funkciju testirati u glavnom programu koji za argument unesen sa tastature ispisuje vrijednosti korijena korištenjem funkcije *“korijen”* kao i korištenjem ugrađene funkcije *“sqrt”* (naravno, rezultati treba da budu isti).