

## STRUKTURE PODATAKA I ALGORITMI – LAB 3

Prije rješavanja zadataka preuzmite na svoje računalo sljedeću datoteku:

- vjezbe3.py

### Zadatak 1:

U datoteci *vjezbe3.py* nalazi se klasa **PogodiBroj** koja nam omogućava igranje igre *Pogodi broj* protiv računala. Instanciranjem ove klase računalo će zamisliti jedan prirodan broj do 100. Klasa ima i jednu metodu (`pokusaj(t)`) koja za zadani broj `t` vraća jednu od sljedećih mogućnosti:

- *Traženi broj je veći* – ukoliko je broj koji je računalo zamislilo veći od `t`
- *Traženi broj je manji* – ukoliko je broj koji je računalo zamislilo manji od `t`
- `[p]` - prirodan broj koji označava da smo broj pogodili u `p`-tom pokušaju.

Primjer koda za igranje igre u interaktivnom sučelju je:

```
>>> igra = PogodiBroj()
>>> igra.pokusaj(80)    ##postavljamo pitanje je li zamišljeni broj 80
'Traženi broj je veći'  ##računalo odgovara da je zamišljeni broj veći
                        od 80
>>> igra.pokusaj(85)    ##postavljamo pitanje je li zamišljeni broj 85
'Traženi broj je manji' ##računalo odgovara da je zamišljeni broj manji
>>> igra.pokusaj(81)    ##postavljamo pitanje je li zamišljeni broj 81
3                        ##računalo odgovara da smo broj pogodili u 3. pokušaju
```

- Osmislite algoritam kako da u najmanjem broju pokušaja pogodite traženi broj. Koji pristup rješavanju problema ćete koristiti (*pohlepni, podijeli pa vladaj, dinamičko programiranje ili backtracking*)?
- Koliko je najviše pokušaja potrebno da bi se pogodio zadani broj (ako se pogađa broj od 1 do 100)? Zašto?
- Napišite program koji će slijedeći osmišljenu strategiju oponašati igranje igre *Pogodi broj* protiv računala (koristeći klasu **PogodiBroj**)

**STRUKTURE PODATAKA I ALGORITMI – LAB 3****Zadatak 2:**

Cijev duljine  $n$  cm potrebno je složiti od manjih komada zadanih duljina  $(d_1, d_2, \dots, d_k)$  na način da se za slaganje cijevi potroši najmanji broj manjih cijevi. Primjerice ako na raspolaganju imamo cijevi duljina: 10, 5, 2 i 1 cm i trebamo složiti cijev duljine 15 cm trebati će nam jedna cijev od 10 cm i jedna cijev od 5 cm (ukupno dvije cijevi).

- Objasnite pohlepni pristup rješavanja ovog problema. Daje li on uvijek točno rješenje? Implementirajte pohlepni pristup za rješenje ovog problema.
- Objasnite dinamički pristup rješenja ovog problema. Implementirajte dinamički pristup rješenja ovog problema.

**Zadatak 3:**

Ispred napuštenog dvorca pronašli smo plan dvorca, koji kaže da se u svakoj prostoriji nalazi po jedan kovčeg s određenim brojem zlatnika. Primjer nacrtu dvorca s pripadnim brojem zlatnika dan je na sljedećoj slici:

30	1	7	15
1	1	12	8
4	5	3	13
1	2	9	11

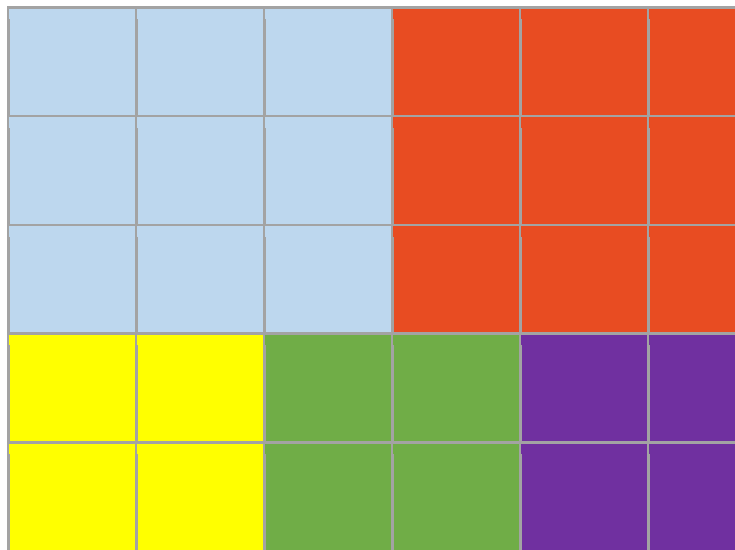
Ući možemo u bilo koju sobu obojenu zelenom bojom u tablici, a izaći u bilo kojoj sobi obojenom plavom bojom u tablici. Cilj nam je proći sobama na način da ukupni broj zlatnika koje ćemo pokupiti bude najveći. Kada smo u nekoj sobi možemo otići samo u sobu koja je u redu ispod (u tablici) i koja se s trenutnom sobom dodiruje lijevom ili desnom donjim vrhom ili donjom stranicom. Primjerice iz sobe u kojoj piše vrijednost 7 možemo otići u sobe u kojima pišu redom vrijednosti: 1, 12 i 8, dok iz sobe u kojoj piše vrijednost 13 možemo otići u sobe u kojima pišu vrijednosti 9 i 11.

- Osmislite algoritam za rješenje ovog problema. Koji pristup rješavanju problema ćete koristiti (*pohlepni, podijeli pa vladaj, dinamičko programiranje ili backtracking*)?
- Implementirajte rješenje za koje ste smislili algoritam.

**Zadatak 4:**

Raspolažemo s čokoladom pravokutnog oblika koja se sastoji od  $n \times m$  manjih kvadratića. Čokoladu je potrebno podijeliti na najmanji broj kvadrata.

Primjerice podjela čokolade dimenzije  $6 \times 5$  na najmanji broj kvadrata dana je na sljedećoj slici (kvadrati su obojeni različitim bojama).



- Osmislite algoritam za rješenje ovog problema. Koji pristup rješavanju problema ćete koristiti (*pohlepni, podijeli pa vladaj, dinamičko programiranje ili backtracking*)?
- Implementirajte rješenje za koje ste smislili algoritam.