

6. Projekti

Tole pravzaprav ni lekcija ampak je skupek problemov, ki jih zastavljamo in rešujemo vsako leto pri predavanjih.

P.1. Tarok

Karte za tarok sestojijo iz navandih kart (barve) in tarokov. Imamo 22 tarokov, označenih z rimskimi številkami od 1 do 22. Običajne štiri barve sta črni barvi križ in pik ter rdeči barvi karo in srce. V vsaki od štirih barv imamo 4 visoke karte: Kralj, Dama, Kaval in Fant ter štiri nizke karte. Pri rdečih so to 1,2,3,4, pri črnih pa 10,9,8,7. Skupaj je torej 54 igralnih kart.

Karte mešamo in delimo tako, da ji gre 6 v talon, preostale pa se v enakih deležih razdelijo igralcem.

Poznamo tarok v treh in v štirih.

Podrobneje si lahko [pravila za tarok ogledate tule](#).

1. Naloga.

Napisali bomo program, ki izpiše vse karte za tarok. Pri tem bomo karte interno predstavili s števili od 0 do vključno 53.

2. Naloga.

Izbrali bomo primerno predstavitev za karte in karte izpisali oz. narisali.

3. Naloga.

Napisali bomo program, ki zmeša in razdeli karte za tarok.

4. Naloga.

Napisali bomo program, ki šteje štihe.

5. Naloga.

Napisali bomo program, ki glede na zaporedje padlih kart na mizi izbere vse karte, ki jih igralec na potezi ima v roki in jih lahko uporabi.

P.2. Sudoku

Namen tega projekta je napisati program, ki rešuje znane uganke sudoku.

Pri [Projektu Euler je 96 naloga](#) namenjena reševanju sudoku. Ker je za začetnika to kar zajeten zalogaj, ga bomo razdelili na manjše dele in bomo pred samim reševanjem problema napisali kup pomožnih programov, s katerimi bomo postopoma izbirali ustrezne podatkovne strukture in algoritme.

1. Naloga.

Naj bo podatek za sudoku niz, dolg 81 znakov, v katerem so vpisane številke, prazna mesta pa so označena z ničlo.

Zgled:

```
"030601080000820001820000000200400700004000900005002006000000028100067000090204030"
```

program naj izpiše sudoku v obliki kvadrata 9 x 9 s poudarjenimi kvadrati 3 x 3.

2. Naloga

Napišite funkcijo `vrstica(sudoku,i)`, ki vrne *i*-to vrstico kvadrata sudoku.

Napišite funkcijo `stolpec(sudoku,i)`, ki vrne *i*-ti stolpec kvadrata sudoku.

Napišite funkcijo `kvadrater(sudoku,i)`, ki vrne *i*-ti 3 x 3 kvadrater kvadrata sudoku.

V vsakem primeru naj bo $0 \leq i \leq 8$, rezultat pa naj bo niz dolžine 9.

3. Naloga

Napišite funkcijo `jePravilnaVrstica(niz)`, ki ugotovi, ali je niz pravilna vrstica nekga sudoku kvadrata. Napišite funkcijo `jePravilen(sudoku)`, ki ugotovi, ali je sudoku kvadrat pravilno delno izpolnjen.

4. Naloga

Napišite funkcijo `kateraVrstica(sudoku,m)`, ki vrne vrstico kvadrata sudoku `m`-tega znak niza. Napišite funkcijo `kateriStolpec(sudoku,m)`, ki vrne `i`-ti stolpec kvadrata Sudoku `m`-tega znak niza. Napišite funkcijo `kateriKadratek(sudoku,m)`, ki vrne `i`-ti 3×3 kvadrata Sudoku `m`-tega znak niza. V vsakem primeru naj bo $0 \leq m \leq 80$, rezultat pa naj bo niz dolžine 9.

5. Naloga

Napišite funkcijo `moznost(sudoku)`, ki vrne seznam množic števil, ki so dopustna.

P.3. CARDIAC

Napisali bi radi simulator za CARDIAC.

1. Naloga.

Vsebino celice pri CARDIACu je predznačena trimestna številka. Predstavimo jo v obliki niza iz štirih znakov. Napišite funkcijo, ki iz te predstavitve naračuna pripadajoče število.

P.4. Generator psevdonaključnih števil

Python ima vgrajeno funkcijo `random()` in nekatere druge z njo povezane funkcije. V dokumentaciji piše, da deluje na podlagi algoritma *Mersenne Twister*.

1. Naloga: Linearni kongruenčni generator.

Za začetek si bomo ogledali linearni kongruenčni generator:

$$x(n+1) = a x(n) + c \text{ mod } m$$

Pri čemer poljubno izberemo $x(0)$.

1. Napisali bomo funkcijo `seed(s)`, ki nastavi vrednost $x(0)$ v odvisnosti od s .

2. Napisali bomo funkcijo `randoml()`, ki ob n -tem klicu vrne "naključno" število $u(n) = x(n)/m$.

Pri tem bomo uporabili globalne spremenljivke. Python globalno spremenljivko prepozna, če jo deklariramo s stavkom `global`.

Problem je pravilna izbira parametrov m , a in c . Ob tem želimo polno periodo.

Knjiga *Numerical recipes* uporablja

$$a = 1664525$$

$$c = 1013904223$$

ter

$$m = 2^{32}$$

P.5. Grafi

Graf $G(V,E)$ sestoji iz množice vozlišč V in množice povezav E , pri čemer E določa simetrično, irefleksivno relacijo \sim na množici V . Če za par vozlišč u in v velja, da je $u \sim v$, pravimo, da sta vozlišči sosednji, neurejen par $e = uv$ pa je povezava s krajiščema u in v .

1. Naloga.

Za poljuben niz sestavite njegov usmerjeni graf v obliki slovarja. Vsak simbol naj dobi seznam naslednikov.

P.6. Bridge

Bridge je znana igra s standardnimi kartami.

1. Naloga.

Izpišite vseh 13 vrednosti kart.

2. Naloga.

Izpišite vse 4 barve kart.

3. Naloga.

Izpišite vseh 52 kart.

4. Naloga.

Zmešajte karte in jih razdelite med igralce.

5. Naloga

Simulirajmo igro, tako da izmed kart, kijih drži v roki igralec, izberemo dopustne karte, tiste torej, ki bi jih v danem trenutku izigrati. Na voljo imamo dva podatka.

- a) Adut: (ena od barv ali igra brez aduta)
- b) Zaporedje kart na mizi (nič, ena, dve ali tri karte)

Za poljubno karto pogledamo, ali bi jo igralec smel izigrati, če bi bil na vrsti in bi jo držal takrat v roki.

6. Naloga

(Nadgradnja 5. naloge) Simulirajmo igro, tako da izmed kart, kijih drži v roki igralec, izberemo dopustne karte, tiste torej, ki bi jih v danem trenutku izigrati. Na voljo imamo tri podatke.

- a) Adut: (ena od barv ali igra brez aduta)
- b) Zaporedje kart na mizi (nič, ena, dve ali tri karte)
- c) Preostale karte, ki jih drži igralec v roki.

Za poljubno karto, ki jo igralec še ima v roki, pogledamo, ali bi jo smel izigrati, če bi bil na vrsti.