

*Uvod v programiranje*  
**Vaje za ponavljanje – 1. del**

Povsod, kjer se ti zdi primerno, napiši svoje *funkcije*, tudi če naloga tega ne zahteva eksplicitno :-)

1. naloga: **Dan v letu**

Napiši program, ki te vpraša po datumu (dan, mesec in leto) in izpiše tekoči dan v tem letu. Pri tem naj upošteva prestopna leta. (“Okrogle stotice” niso prestopna<sup>1</sup> leta, razen če so deljiva s 400.)

Primer:

```
Vnesi dan v mesecu: 5
Vnesi mesec v letu: 2
Vnesi leto: 2008
```

5. 2. 2008 je 36. dan v letu.

2. naloga: **Strešniki**

Radi bi na novo prekrili streho in nas zanima, koliko strešnikov potrebujemo za to. Napravi program, ki mu vnesemo dolžino in širino strehe in dolžino in širino strešnikov, vse v centimetrih. Program naj izračuna, koliko strešnikov potrebujemo. (Stranice strešnikov morajo biti vzporedne stranicam strehe. Upoštevati je potrebno, da na robu odrezanih strešnikov ne moremo ponovno uporabiti. Strešnike lahko postavljamo bodisi pokončno bodisi ležeče, vsi pa morajo biti obrnjeni v isto smer.)

Primer:

```
Vnesi dolžino strehe: 5
Vnesi širino strehe: 3
Vnesi dolžino strešnika: 2
Vnesi širino strešnika: 2
```

Potrebujemo 6 strešnikov.

3. naloga: **Ključavnica**

Na ponovoletni razprodaji imajo ugodne kolesarske ključavnice na številke. Številke od 0 do 9 so na okroglih številčnicah, ki so prosto vrtljive.

Vsaka ključavnica ima 4 številčnice na klik, kar pomeni, da se sliši vsak premik katere koli izmed štirih številčnic in sicer po en klik za vsako spremembo za eno mesto. Tako na primer sprememba številčnice z 1 na 3 povzroči 2 klika.

Napravi program, ki zna izračunati koliko je najmanjše število klikov na ključavnici, če želiš preiti iz ene na drugo kombinacijo številke na ključavnici.

Primer:

```
Vnesi začetno številko: 2222
Vnesi končno številko: 3131
```

Število klikov: 4

---

<sup>1</sup>Glej: [http://en.wikipedia.org/wiki/Leap\\_year](http://en.wikipedia.org/wiki/Leap_year)

#### 4. naloga: **Televizor**

Pri nakupih televizorjev je večinoma podan podatek o dolžini diagonale televizorja. Iz dolžine diagonale in razmerja stranic televizorja želimo izračunati njegove stranice.

Sestavi program, ki prebere:

- dolžino diagonale,
- razmerje višine in širine televizorja,

in izračuna višino in širino televizorja. Obe dolžini naj program izpiše zaokroženi na najbližje celo število.

Primer:

```
Vnesi dolžino diagonale: 52
Vnesi razmerje med stranicama.
Koliko enot meri višina? 9
Koliko enot meri širina? 16
```

```
Višina televizorja: 25
Širina televizorja: 45
```

#### 5. naloga: **Števke, števke, števke**

Napiši program, ki izpiše vsa tista štirimestna števila, ki imajo razliko prvih dveh števk enako vsoti drugih dveh števk.

Eno tako število je npr. 7313, ker je  $7 - 3 = 1 + 3 = 4$ .

#### 6. naloga: **Knjižna dieta**

Tudi knjižni molji morajo jesti kosilo. Molj Mirko je še posebej izbirčen in hoče vsak dan jesti natanko isto: 6 leksikonov in za sladico še 5 stripov. Tudi če ima na razpolago na primer 12 leksikonov in 1000 stripov (ali pa obratno), mu bo to zadoščalo samo za dve kosili, vse ostale stripe bo pustil pri miru in rajši izstradal.

Napiši program, ki prebere, koliko je na knjižni polici leksikonov in koliko stripov (vsako število v svoji vrstici), nato pa izpiše, za koliko Mirkovih kosil bodo te knjige zadoščale.

Primer:

```
Koliko leksikonov je na knjižni polici? 20
Koliko stripov je na knjižni polici? 27
```

```
Število kosil: 3
```

#### 7. naloga: **Fibonacci**

Napiši program, ki izračuna vsoto vseh sodih členov Fibonaccijevega<sup>2</sup> zaporedja, ki ne presegajo 1.000.000. Prvi in drugi člen Fibonaccijevega zaporedja sta 1, vse ostale pa dobimo tako, da seštejemo prejšnja dva. Prvih nekaj členov zaporedja:

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

---

<sup>2</sup>Glej: [http://en.wikipedia.org/wiki/Fibonacci\\_number](http://en.wikipedia.org/wiki/Fibonacci_number)

### 8. naloga: **Kašča putke Amalije**

Putka Amalija se pripravlja na dolgo zimo. Vsak dan v kaščo spravi tri zrna koruze. Napiši program, ki prebere, koliko dni je Amalija nabirala zrna, nato pa nariše zalogo v njeni kašči, kakor vidiš na spodnjem primeru. Za vsako zrno izpiši eno črko "o", ograjice pa so iz znakov "#". Kašča je vedno široka 10 enot, višina pa se prilagodi količini koruze.

Primer:

Koliko dni je Amalija nabirala zrnje? 9

```
#####  
#oooooooo #  
#ooooooooo#  
#ooooooooo#  
#####
```

### 9. naloga: **Žoganje**

Na FMF so se vsi bruci postavili v krog in se začeli igrati z žogo. Igra poteka tako, da si žogo preprosto podajajo v krog. Žogo ima na začetku v rokah vedno prvi igralec, ki jo poda drugemu, ta jo poda tretjemu in tako naprej. Ker se igra v krogu, poda zadnji prvemu.

Napiši program, ki prebere število igralcev in število podaj, ter izpiše kateri igralec bo imel žogo po toliko podajah.

Primer<sup>3</sup>:

Število igralcev: 4

Število podaj: 10

Na koncu ima žogo 3. igralec.

### 10. naloga: **Solatiča**

Gospod in gospa Solatič sta zelo ponosna na svojo gredico solate. Glave solate so vse enake, posajene v eni ravni vrsti in enakomerno medsebojno razmaknjene. Če začne padati toča, se postavita s svojimi dežniki med glave solate, da jih čim več zaščitita. Napiši program, ki prebere štiri števila (vsako v svoji vrsti) in pove, koliko glav solate sta Solatiča uspela zaščititi pred točo.

Prvi par števil opisuje gospodov dežnik. Prva številka je premer dežnika, druga številka pa pove, katera je najbolj leva glava, ki jo dežnik pokriva. Drugi par števil opisuje gospejin dežnik.

Primer<sup>4</sup>:

Velikost dežnika gospoda: 3

Začetek dežnika gospoda: 2

Velikost dežnika gospe: 5

Začetek dežnika gospe: 3

Število zaščitениh glav: 6

<sup>3</sup>Če igrajo štirje igralci, bo v prvi podaji 1. igralec podal drugemu, v drugi podaji bo 2. igralec podal tretjemu. Po desetih podajah bo žoga naredila dva kroga in končala pri tretjemu igralcu.

<sup>4</sup>Dežnika se nad 3. in 4. glavo prekrivata, zato so zaščitene le glave številka 2 do 7.

### 11. naloga: **Luknjasta**

Napiši program, ki prebere neko število in nato izpiše, koliko je lukenj v tem številu. Z luknjami so mišljene luknje v vsaki posamezni številki: številke 0, 4, 6 in 9 imajo po eno luknjo, številka 8 ima dve, preostale številke pa nimajo lukenj.

Primer<sup>5</sup>:

Vnesi naravno število: 5423087  
Število 5423087 ima 4 lukenj.

### 12. naloga: **Fußball**

Dekan se je odločil, da bo za svoje študente dal zgraditi čisto pravo nogometno igrišče. Dolžina in širina pravega nogometnega igrišča morata biti v razmerju 2 : 1. Igrišče, ki je dolgo 50 m in široko 25 m, je torej pravilno nogometno igrišče. Travnik, na katerem bodo naredili igrišče, je pravokotne oblike in je dolg  $m$  metrov in širok  $n$  metrov ( $m$  in  $n$  sta naravni števili med 1 in 100). Robovi igrišča bodo vzporedni z robovi travnika, postavljeno pa bo lahko po dolžini ali širini. Za igrišče pa velja še to, da ne sme biti daljše od 80 m in krajše od 30 m.

Tvoja naloga je napisati program, ki bo glede na dimenziji travnika izračunal dolžino in širino največjega igrišča, ki ga je možno zgraditi pod zgornjimi pogoji. Program naj prebere  $m$  in  $n$ . Izpiše naj dimenzije največjega mogočega igrišča. Vse mere so v metrih, računamo do centimetrov natančno. Če je prostora več kot sme biti velikost igrišča, potem naj program vrne največjo še dovoljeno velikost. Če je površina neustrezna, naj program izpiše: "Igrišča žal ni mogoče zgraditi!"

Primer #1:

Dolžina travnika: 100.0  
Širina travnika: 10.0

Igrišča žal ni mogoče zgraditi!

Primer #2:

Dolžina travnika: 35.0  
Širina travnika: 51.0

Velikost igrišča: 51.00 x 25.50

### 13. naloga: **Požri ničle**

Putke so znane po tem, da nikakor ne prenesejo ničel. Zato vedno, ko dobijo kakšno število, iz njega izključujejo vse ničle. Število 200703 bi putke nemudoma spremenile v 273. Napiši program, ki bo s standardnega vhoda prebral začetno število  $n \in \mathbb{N}$  in izpisal število, ki ga putke dobijo po tem, ko požrejo vse ničle.

Primer:

Vnesi število: 5047006

Po obdelavi ostane: 5476

---

<sup>5</sup>Ena luknja je v številki 4, ena v 0 in dve v 8, kar da skupaj 4 luknje.

#### 14. naloga: **Kokošja farma**

Na kokošji farmi so ravnokar kupili zalogo hrane. Ker pa pred nakupom niso naredili nobenih izračunov, ne vedo, za koliko časa bo ta zaloga zadostovala. Kokoši pojedjo prvi teden 1 enoto hrane; ker se malo poredijo, pojedjo drugi teden 2 enoti hrane; tretji teden pojedjo 3 enote hrane in tako naprej – vsak naslednji teden pojedjo za eno enoto hrane več kot prejšnji teden.

Napiši program, ki bo kmetu Francu izračunal, za koliko tednov bo zadostovala njegova zaloga hrane.

Primer<sup>6</sup>:

Vnesi količino nabavljene hrane: 11

Ta količina zadostuje za 4 dni.

#### 15. naloga: **Bum!**

Bruci matematike so utrjevali poštevanko in se igrali igrico “Bum!”. Igra poteka tako, da v krogu štejejo od 1 dalje tako, da vsak pove naslednje večje število. Izjema so mnogokratniki števila 3 in pa tista števila, ki vsebujejo števko 3. Tedaj študent, ki je na vrsti, namesto števila zavpije: “Bum!” Napravi program, ki bo brucom pomagal pravilno šteti pri tej igrici.

Primer:

Do kod naj štejem? 13

1

2

Bum!

4

5

Bum!

7

8

Bum!

10

11

Bum!

Bum!

#### 16. naloga: **Duckling**

Račke sicer prav dobro obvladajo matematiko, vendar si jo predstavljajo malo po svoje. Uporabljajo sicer normalni desetiški številski sistem, le da so številke zamenjale z črkami. Putke so našle napis, s katerim menijo, da lahko razvozljajo njihovo skrivnost zapisovanja števil in te prosijo, da jim pri tem pomagaš.

Napravi program, ki reši kriptaritem, pri katerem moraš najti različne številke, tako, da bo izpolnjena račja enačba:

$$PEEP * PEEP = DUCKLING$$

Nalogo reši tako, da program izpiše enačbo z ustreznimi številkami.

---

<sup>6</sup>V prvih štirih tednih bodo kokoši pojedle  $1 + 2 + 3 + 4 = 10$  enot hrane, preostala 1 enota pa ni dovolj za peti teden.

### 17. naloga: **Ljudožerska**

Pleme ljudožercev je bilo na zadnjem pohodu precej uspešno: zajeli so  $n$  prijetno tolstih pripadnikov sovražnega plemena. Še dobro; zaloge hrane so jim namreč ravnokar pošle. Zdaj jih zanima, koliko časa bo trajala nova zaloga.

Da se nasitijo, vsak dan skuhajo 7 ljudi. Ker pa so vraževerni, v primeru, da jim po tistem ostane  $13k$  ljudi (za nek  $k \in \mathbb{N}$ ), v kotel hitro vtaknejo še enega.

Sestavi program, ki prebere število  $n$  in izpiše, za koliko dni zadoščajo zaloge. Če jim na koncu ostane 6 ali manj ujetnikov, to ne zadošča za še en obrok.

Primer #1:

```
Vnesi število ujetnikov: 123
Zaloge zadoščajo za 17 dni.
```

Primer #2:

```
Vnesi število ujetnikov: 91
Zaloge zadoščajo za 13 dni.
```

### 18. naloga: **Podčrtano**

V časopisih, knjigah in dokumentih so naslovi običajno izpisani bolj vpadljivo, da jih hitreje opazimo. Danes tega ni težko doseči: za naslov izberemo večjo in bolj odebeljeno pisavo, pa je. Včasih pa, ko so ljudje večino dokumentov natipkali s pisalnim strojem, ki je seveda imel eno samo pisavo, to ni bilo tako preprosto. En način, kako naredimo naslov bolj vpadljiv brez uporabe dodatnih pisav, je, da med njegovimi črkami izpišemo dodatne presledke. Na primer naslov "Putke za gripo krivijo prašiče" bi zapisali takole:

```
P u t k e   z a   g r i p o   k r i v i j o   p r a š i č e
```

Vidimo, da pri tem presledke iz naslova obravnavamo kot vse ostale znake – tudi okrog njih smo vrinili nove presledke, enako kot okrog črk. No, to še ni zelo vpadljivo. Lahko pa naslov še dodatno poudarimo in ga "podčrtamo" tako, da v naslednji vrstici izpišemo toliko enačaje (znakov "="), da sežejo od prvega točno do zadnjega znaka naslova. Takole:

```
P u t k e   z a   g r i p o   k r i v i j o   p r a š i č e
=====
```

Napiši program, ki prebere naslov in izpiše njegovo poudarjeno (razmaknjeno in podčrtano) različico.

Zgled:

```
Vnesi naslov: Katastrofa!
```

```
K a t a s t r o f a !
=====
```

### 19. naloga: **projecteuler.net**

Obišči spletno stran <http://projecteuler.net/>, na kateri je več kot 200 matematičnih problemov, ki so mišljeni za reševanje z računalnikom. Lahko se jih lotiš kar v Pythonu ali pa uporabiš kakšno drugo orodje.

Registriraj se na tej strani in poskusi rešiti čimveč nalog. Začneš lahko z lažjimi nalogami, npr. #1, #3, #4, #5, #6, ...