

# Uvod v programiranje, 2008/09

## 1. kolokvij, skupina B

1. Sestavi funkcijo, ki preveri, ali je dani znesek mogoče plačati z ENAKIM številom kovancev različnih vrst. Za parametre naj funkcija dobi znesek, ki ga želimo plačati, ter vrednosti treh vrst kovancev. Če je znesek mogoče plačati le z eno vrsto kovanca, naj funkcija vrne vrednost 1. Če je znesek mogoče plačati z dvema vrstama kovancev (npr. 2 kovanca a in 2 kovanca b), naj funkcija vrne 2, če pa je znesek mogoče plačati le z uporabo vseh treh vrst kovancev (npr. 4 kovanci a, 4 kovanci b in 4 kovanci c), pa naj vrne vrednost 3. Če zneska po tem pravilu ni mogoče plačati, naj funkcija vrne False.

```
>>> kovanci(1243, 1, 5, 10)
1
>>> kovanci(1243, 2, 5, 9)
2
>>> kovanci(1243, 2, 4, 5)
3
>>> kovanci(17, 3, 4, 2)
False
```

2. Sestavi funkcijo, ki za dano naravno število vrne vsoto vsake druge številke, torej vsoto prve, tretje, pete, ... številke, pri čemer jih štejemo od leve proti desni. Nalogo lahko rešiš brez seznamov, ali pa s pomožnim seznamom števk.

```
>>> vsotaStevk(1234567)
16
>>> vsotaStevk(12345678)
16
>>> vsotaStevk(0)
0
```

3. Sestavi funkcijo, ki za parametre dobi podatke o začetku varčevanja (leto, mesec in znesek) ter seznam gibanja mesečnih obrestnih mer. Funkcija naj vrne leto in mesec (seznam z dvema elementoma), ko je bilo na varčevalnem računu največ denarja. Pomagaj si s pomožnim seznamom, v katerega naračunaš stanja na varčevalnem računu po posameznih mesecih, nato pa poiščeš, na katerem mestu se nahaja največji znesek.

```
>>> obresti(2004, 10, 500, [0.05, 0.01, 0, -0.1, -0.1, 0.2, 0.1, -0.03])
[2005, 5]
```

4. Sestavi funkcijo, ki dobi za parametra dva seznama, ter zgradi in vrne matriko, ki ima po glavni diagonali elemente prvega seznama, nad glavno diagonalo pa elemente drugega seznama. Povsod drugje v matriki naj bodo ničle. Drugi seznam mora biti za en element krajši od prvega (če ni, naj funkcija vrne False). Uporabi gnezdeni izpeljani seznam.

```
>>> bidiagonalna([1, 2, 3], [4, 5])
[[1, 4, 0], [0, 2, 5], [0, 0, 3]]
>>> bidiagonalna([1, 2, 3, 4], [5, 6, 7])
[[1, 5, 0, 0], [0, 2, 6, 0], [0, 0, 3, 7], [0, 0, 0, 4]]
>>> bidiagonalna([1, 2, 3], [4, 5, 6, 7])
False
```