

## Programiranje I: 1. izpit

3. februar 2011

Čas reševanja je 120 minut. Veliko uspeha!

### 1. naloga (25 točk)

Pri programiranju obstaja več pristopov k zapisu dolgih imen spremenljivk. Nekateri pišejo `dolgoImeSpremenljivke`, drugi pa `dolgo_ime_spremenljivke`.

**a)** Sestavite funkcijo `pobrisiPodcrtaje`, ki sprejme ime spremenljivke s podčrtaji in malimi začetnicami ter vrne pripadajoče ime brez podčrtajev in z velikimi začetnicami. Primer uporabe:

```
>>> pobrisiPodcrtaje("dolgo_ime_spremenljivke")
"dolgoImeSpremenljivke"
```

**b)** Sestavite funkcijo `dodaj_podcrtaje`, ki deluje ravno obratno kot `pobrisiPodcrtaje`. Primer uporabe:

```
>>> dodaj_podcrtaje("dolgoImeSpremenljivke")
"dolgo_ime_spremenljivke"
```

### 2. naloga (25 točk)

*Izbor reda  $n$*  je strogo naraščajoče zaporedje naravnih števil, ki so vsa med 1 in  $n$ . Na primer,  $[1, 2, 3, 7]$  je izbor reda 9 in *ni* izbor reda 5. Vsi izbori reda 2 so  $[], [1], [2]$ , in  $[1, 2]$ . Sestavite generator `izbori(n)`, ki zgenerira vse izbore reda  $n$ .

### 3. naloga (25 točk)

Na spletni učilnici najdete datoteko `IskalnoDrevo.py`, v kateri je dan razred `iskalnih dreves`.

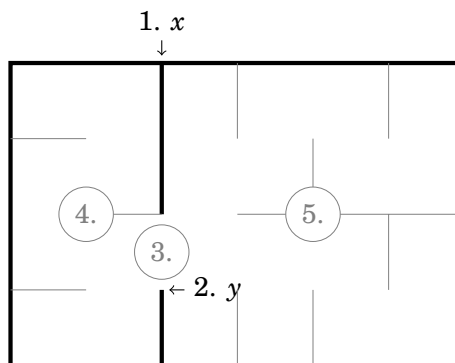
**a)** Razredu dodajte metodo `predhodnik(self, x)`, ki vrne največji element v drevesu `self`, ki je še strogo manjši od  $x$ . Na primer, če so v drevesu `d` shranjena števila 1, 3, 5, 10, 12, potem ukaz `d.predhodnik(9)` vrne 5.

**b) (+10 točk)** Sestavite še metodo `interval(self, x, y)`, ki vrne seznam tistih elementov v drevesu, ki so strogo večji od  $x$  in strogo manjši od  $y$ .

#### 4. naloga (25 točk)

Iz pravokotnika velikosti  $v \times s$  lahko ustvarimo naključen labirint po naslednjem postopku:

- Če je bodisi višina  $v$  bodisi širina  $s$  enaka 1, končamo.
- Če je višina  $v$  manjša od širine  $s$ :
  1. Izberemo naključen  $x \in \{1, \dots, s-1\}$ .
  2. Izberemo naključen  $y \in \{0, \dots, v-1\}$ .
  3. Pravokotnik predelimo na pol s črtama od točke  $(x, 0)$  do točke  $(x, y)$  ter od točke  $(x, y+1)$  do točke  $(x, v)$ .
  4. Postopek ponovimo na levi polovici.
  5. Postopek ponovimo na desni polovici.



- Če je višina  $v$  ni manjša od širine  $s$ , ravnamo podobno, le da pravokotnik predelimo po višini.

V *Mathematici* sestavite funkcijo `labirint[v_, s_]`, ki po zgornjem postopku sestavi naključen labirint iz ustreznih grafičnih ukazov.