

Programiranje I: 1. izpit

19. januar 2022

Čas reševanja je 120 minut. Veliko uspeha!

1. naloga

a) Napišite predikat `sta_pravokotna : int * int * int -> int * int * int -> bool`, ki pove, ali sta dva vektorja pravokotna. Vektorje predstavimo s trojicami celih števil.

b) Napišite funkcijo `postkompozicija : ('a -> 'b) -> ('b -> 'c) -> 'a -> 'c`, ki sprejme dve funkciji in vrne njun kompozitum.

```
# let deli_z m n = n / m;;
# postkompozicija ((+) 82) (deli_z 82) 3362;;
- : int = 42
# postkompozicija (deli_z 82) ((+) 82) 3362;;
- : int = 123
```

c) Napišite funkcijo `dopolni : 'a -> 'a option list -> 'a list`, ki sprejme privzeto vrednost in seznam morebitnih vrednosti ter vrne seznam, v katerem namesto manjkajočih vrednosti nastopa privzeta, obstoječe vrednosti pa ostanejo nespremenjene. Za vse točke naj bo funkcija repno rekurzivna.

```
# dopolni 3 [Some 1; None; Some 2];;
- : int list = [1; 3; 2]
```

d) Naravno število v drugačni bazi predstavimo s seznamom celih števil, kjer so na zadnjem mestu enice, na predzadnjem "desetice" in tako naprej.

Napišite funkcijo `pretvori : int -> int list -> int`, ki sprejme seznam pozitivnih celih števil in bazo ter vrne podano število v desetiškem sistemu kot celo število.

```
# pretvori 8 [6; 4; 4; 2];; (* 6*8^3 + 4*8^2 + 4*8 + 2 *)
- : int = 3362
```

2. naloga

V datotečnih sistemih, sistemih za nadzor različic in kriptovalutah za učinkovito zagotavljanje pristnosti podatkov uporabljamo *Merkleova drevesa*. To so običajna dvojiška drevesa, le da vsako vozlišče poleg podatka in otrok hrani še svojo *zgostitev* (angleško *hash*), ki je celo število, ki tvori nekakšen podpis celotne vsebine drevesa.

```
type 'a merkle = List | Vozlisce of 'a vozlisce
and 'a vozlisce = {
  levo : 'a merkle;
  podatek : 'a;
  desno : 'a merkle;
  zgostitev : int;
}
```

Zgostitev drevesa izračunamo z zgostitvenimi funkcijami, ki sprejmejo zgostitev levega otroka, podatek v vozlišču in zgostitev desnega otroka:

```
type 'a zgostitev = int -> 'a -> int -> int
```

Za zgostitev praznega drevesa izberemo število 0.

Na primer, za zgostitveno funkcijo:

```
let primer_h l p d = ((l * 3) + (p * 5) + (d * 7)) mod 11
```

je spodnje drevo veljavno Merkleovo drevo:

```
let drevo : int merkle = Vozlisce {
  levo = Vozlisce {
    levo = Vozlisce { levo = List; podatek = 10; desno = List; zgostitev = 6 };
    podatek = 14;
    desno = Vozlisce { levo = List; podatek = 474; desno = List; zgostitev = 5 };
    zgostitev = 2;
  };
  podatek = 57;
  desno = Vozlisce {
    levo = List;
    podatek = 12;
    desno = Vozlisce { levo = List; podatek = 513; desno = List; zgostitev = 2 };
    zgostitev = 8;
  };
  zgostitev = 6;
}
```

a) Napišite predikat `preveri : 'a zgostitev -> 'a merkle -> bool`, ki sprejme zgostitveno funkcijo in drevo ter preveri, ali so vse zgostitve v drevesu pravilno izračunane.

b) Napišite funkcijo `prestej_napacne : 'a zgostitev -> 'a merkle -> int`, ki vzame zgostitveno funkcijo in drevo ter vrne število napačno poračunanih zgostitev v drevesu. Zgostitev vozlišča je napačna, če ne ustreza definiciji računanja zgostitve za vozlišče, ne glede na to, ali sta zgostitvi otrok pravilni ali ne.

c) Napišite funkcijo `popravi : 'a zgostitev -> 'a merkle -> 'a merkle`, ki sprejme (potencialno napačno) Merkleovo drevo in vrne drevo iste oblike z istimi podatki, le da so vse zgostitve v njem pravilno izračunane.

3. naloga

Radko Razkurnik je poklicni internetni trol, ki ljudi s pogostim menjanjem stališč zapenja v ne-konstruktivne debate. Glede aktualnega družbenega dogajanja zna zagovarjati n različnih stališč in vsakič, ko preklopi iz stališča i v j , s svojo nedoslednostjo spodbudi k_{ij} komentarjev. (Seveda nekaj komentarjev sproži tudi z vztrajanjem pri stališču i). Dan je kratek, zato lahko Radko napiše največ m komentarjev, pri čemer je m precej večji od n , saj Radko hitreje piše kot misli. Kako naj izbere svoja stališča s_1, s_2, \dots, s_m , da bo skupaj sprožil največ komentarjev, kar jih lahko?

Na primer pri matriki

$$k = \begin{pmatrix} 5 & 25 & 3 & 2 \\ 1 & 4 & 25 & 1 \\ 2 & 1 & 3 & 2 \\ 4 & 1 & 40 & 5 \end{pmatrix}$$

so optimalne vsote pri $m = 1, 2, 4, 6, 9$ enake:

$$k_{43} = 4$$

$$k_{12} + k_{23} = 50$$

$$k_{12} + k_{23} + k_{34} + k_{43} = 92$$

$$k_{12} + k_{23} + k_{34} + k_{43} + k_{34} + k_{43} = 134$$

$$k_{43} + k_{34} + k_{43} + k_{34} + k_{43} + k_{34} + k_{43} + k_{34} + k_{43} = 208$$

Napišite čim bolj učinkovito funkcijo `trololo`, ki sprejme matriko $(k_{ij})_{1 \leq i, j \leq n} \in \mathbb{N}^{n \times n}$ ter $m \in \mathbb{N}$ in izračuna zgornji seznam. *Nalogo lahko rešujete v Pythonu ali OCamlu.*