Programiranje I: 1. izpit

24. januar 2019

Čas reševanja je 150 minut. Veliko uspeha!

1. naloga

a) Napišite funkcijo

```
podvoji_vsoto : int -> int -> int,
```

ki za dani števili izračuna dvakratnik njune vsote.

b) Napišite funkcijo

```
povsod_vecji : 'a * 'b * 'c -> 'a * 'b * 'c -> bool,
```

ki sprejme dve trojici, in preveri, ali ima prva trojica na vseh komponentah večje vrednosti.

c) Napišite funkcijo

```
uporabi_ce_lahko : ('a -> 'b) -> 'a option -> 'b option,
```

ki sprejme funkcijo in element tipa `option` ter v primeru, da element vsebuje vrednost, uporabi funkcijo na vrednosti, sicer vrne `None`.

d) Napišite funkcijo

```
pojavi_dvakrat : 'a -> 'a list -> bool,
```

ki preveri ali se element v seznamu pojavi natanko dvakrat.

e) Napišite funkcijo

```
izracunaj_v_tocki : 'a -> ('a -> 'b) list -> 'b list,
```

ki sprejme točko in seznam funkcij ter vrne seznam rezultatov, ki jih dobimo, ko funkcije izračunamo v točki. Funkcija naj bo repno rekurzivna.

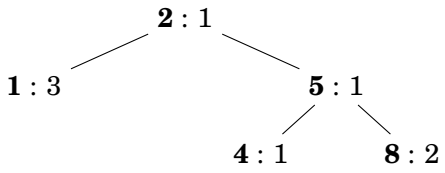
f) Napišite funkcijo

```
eksponent : int -> int -> int,
```

kjer `eksponent x p` izračuna vrednost x^p. Funkcija naj bo repno rekurzivna, kar lahko preverite z izračunom števila $1^{1.000.000}$.

2. naloga

Podatkovno strukturo dvojiških iskalnih dreves pogosto uporabljamo za predstavitev množic. V tej nalogi bomo drevesa uporabljali za predstavitev multi-množic, kjer se lahko isti element pojavi večkrat, npr. $\{1,1,2\}$ sedaj ni več enaka kot $\{1,2\}$. Da bi se izognili podvajanju vozlišč, sedaj vsako vozlišče dodatno nosi še števec za pojavitve elementa, ohranjamo pa strukturo dvojiških iskalnih dreves. Na primer multi-množico $\{2,5,1,4,1,1,2,8,8\}$ bi lahko predstavili z:



a) Napišite tip `'a mm_drevo`, ki vsebuje prazno drevo, in pa drevesa zgrajena iz vozlišč, ki vsebujejo element tipa `'a`, števec tipa `int` in pa levo in desno poddrevo.

b) Napišite funkcijo `vstavi : 'a mm_drevo -> 'a -> 'a mm_drevo`, ki dani element vstavi v multi-množico. Pri tem naj doda novo vozlišče, če elementa še ni v drevesu, sicer pa poveča primeren števec. Drevo naj ohrani strukturo dvojiškega iskalnega drevesa.

c) Napišite funkcijo `multimnozica_iz_seznama : 'a list -> 'a mm_drevo`, ki sestavi drevo, ki predstavlja multi-množico, ki vsebuje elemente danega seznama.
Namig: Funkcijo lahko uporabite, da ustvarite testne primere.

d) Napišite funkcijo `velikost_multimnozice : 'a mm_drevo -> int`, ki vrne velikost multi-množice, ki jo drevo predstavlja.

e) Napišite funkcijo `seznam_iz_multimnozice : 'a mm_drevo -> 'a list`, ki vrne seznam, ki vsebuje vse elemente multi-množice. Za vse točke naj bo vrnjen seznam urejen, pri tem pa ni dovoljeno uporabiti urejevalnih funkcij.
DODATNA NALOGA: Za dosego dodatnih točk naj bo funkcija v celoti repno rekurzivna (ne pozabite preveriti katere vgrajene funkcije na seznamih so repno rekurzivne).

3. naloga

Nalogo lahko rešujete v Pythonu ali OCamlu.

Žabica se je izgubila v močvari in želi kar se da hitro odskakljati ven. Na srečo močvara vsebuje veliko muh, s katerimi si lahko povrne energijo, kajti utrujena žabica ne skoči daleč.

Želimo ugotoviti, kako hitro lahko žabica odskaklja iz močvare. Močvaro predstavimo z tabelo, kjer žabica prične na ničtem polju. Če je močvara dolžine k, je cilj žabice priskakljati vsaj na k-to polje ali dlje (torej prvo polje, ki ni več vsebovano v tabeli).

Energičnost žabice predstavimo z dolžino najdaljšega možnega skoka. Torej lahko žabica z količino energije e skoči naprej za katerokoli razdaljo med 1 in e, in če skoči naprej za k mest ima sedaj zgolj $e-k$ energije. Na vsakem polju močvare prav tako označimo, koliko energije si žabica povrne, ko pristane na polju. Tako se včasih žabici splača skočiti manj daleč, da pristane na polju z več muhami. Predpostavimo, da ima vsako polje vrednost vsaj 1, da lahko žabica v vsakem primeru skoči naprej.

V primeru `[2, 4, 1, 2, 1, 3, 1, 1, 5]` lahko žabica odskaklja iz močvare v treh skokih, v močvari `[4, 1, 8, 2, 11, 1, 1, 1, 1, 1]` pa potrebuje zgolj dva.

Za vse točke naj bo funkcija primerno memoizirana. Funkcija mora v nekaj sekundah izračunati rešitev za močvirje dolžine 50, ki ima na vsakem polju število 10.