

# Programiranje II

2018/19

## 1. izpit, FAMNIT

Izpit rešujete posamično. Naloge so enakovredne. Pri reševanju je dovoljena uporaba literature. Čas pisanja izpita je 90 minut.

Veliko uspeha!

IME IN PRIIMEK:	
VPISNA ŠTEVILKA:	
ŠTUD. POGRAM:	
PODPIS:	

**Naloga 1** (25%)

Napiši funkcijo `count : int list -> int list`, kjer je prvi parameter seznam celih števil `int list`. Rezultat funkcije `count` je seznam celih števil, kjer je `i`-ti element vsota vseh predhodnikov.

**Naloga 2 (25%)**

Dan je parametriziran tip `'a kv_array`, ki je definiran kot polje parov ključ/vrednost. Prva komponenta para je *ključ* tipa `int` in druga komponenta je *vrednost* tipa `'a`.

```
type 'a kv_array = int*'a array
```

Napiši polimorfično funkcijo

```
stik : ('a->'a->'a) -> 'a kv_array -> 'a kv_array -> 'a kv_array,
```

ki naredi stik dveh polj parov ključ/vrednost na naslednji način. Ključi vhodnih dveh polj so neurejeni !

Imenujmo prvi parameter  $f$ , drugi  $a_1$ , tretji  $a_2$  in rezultat  $a_3$ . Za vsak element  $(k, v)$  iz polja  $a_3$  obstaja par  $(k, v_1)$  iz  $a_1$  in par  $(k, v_2)$  iz  $a_2$  tako, da velja  $v = f\ v_1\ v_2$ .

*Razlaga:* Polje  $a_3$  (rezultat) vsebuje pare s ključi, ki se nahajajo v parih polj  $a_1$  in  $a_2$  (v obeh!). Z uporabo funkcije  $f$  združimo vrednosti parov iz polj  $a_1$  in  $a_2$ , ki imajo enake ključe.

**Naloga 3** (25%)

a) Definiraj parametriziran tip vozlišč drevesa z imenom `'a` `g``rm`, ki imajo lahko nič vej (nobenega otroka), eno vejo (enega otroka) ali dve veji (dva otroka). Vsako vozlišče vsebuje ključ tipa `'a`.

Za definicijo tipa `'a` `g``rm` uporabi unijo tipov.

b) Kreiraj primerek tipa `'a` `g``rm`, kjer je `'a=string` in vsebuje vsaj 3 vozlišča!

c) Napiši funkcijo `izpis` : `'a` `g``rm` `->` `unit`, ki izpiše ključne vozlišče drevesa po principu najprej-v-globino.

**Naloga 4 (25%)**

Definiraj razred `kv_class`, ki predstavlja shrambo parov ključ/vrednost. Ključ je tipa `int` in vrednost je tipa `string`.

Razred naj vsebuje naslednji metodi.

- Metoda `put : int*string -> unit` vstavi ključ in vrednost v `kv_class`.
- Metoda `get : int -> string`, ki za dani ključ vrne pripadajočo vrednost.

Predpostaviš lahko, da dodanih parov ne brišemo.

*Nasvet:* Uporabi čim bolj enostavno podatkovno strukturo za shranjevanje parov. Na primer, za shrambo parov ključ/vrednost lahko uporabiš enostaven seznam parov.