

Algoritmi in podatkovne strukture 1

6. februar 2025

Ime in priimek: _____

Vpisna številka: _____

Pri reševanju si lahko pomagate z enim A4 listom zapiskov.

Svoje odgovore pišite jasno in jih utemeljite.

Ocenjuje se odgovore na izpitni poli. Dodatni listi so namenjeni pomoči pri reševanju in jih ne oddate.

Čas reševanja: 90 min.

Naloga:	1	2	3	4	Skupaj
Točke:	25	25	25	25	100
Ocena:					

1. Čudežnemu drevesu poganjajo listi. Na začetku vsakega leta požene en list, ki se na drevesu obdrži t_i let. Če npr. list požene na začetku tretjega leta in se obdrži $t_3 = 2$ leti, bo prisoten na drevesu v tretjem in četrtem letu našega štetja. Botaniki so nam pripravili seznam vrednosti t_i za n let ($i = 1 \dots n$), prosijo pa nas, da jim učinkovito izračunamo, koliko listov bo na drevesu v vsakem izmed prvih n let.

[15] (a) Opišite čim bolj učinkovit algoritem, ki izračuna odgovor na zastavljeno vprašanje. Navedite in utemeljite tudi njegovo časovno in prostorsko zahtevnost.

[10] (b) V C++ implementirajte funkcijo `listi`, ki bo sprejela seznam števil t_i in vrnila enako dolg seznam, ki vsebuje število listov na drevesu za vsako leto.

Implementacija lahko ustreza manj učinkovitemu postopku od prej opisanega, vendar bo zato prejela manj točk (v tem primeru jo tudi na kratko opišite). Pri implementaciji lahko uporabljate vso funkcionalnost standardne knjižnice C++.

```
1 vector<int> t = {3,1,3,2,1,3,4};  
2 vector<int> l = listi(t); // {1,2,2,2,3,1,2}
```

2. Dijkstrov algoritem za splošne redke grafe (z uporabo kopice) bomo izvajali na različnih družinah grafov z uteženimi povezavami. Za vsako izmed spodnjih družin navedite in utemeljite časovno zahtevnost izvajanja algoritma. Pri oceni zahtevnosti bodite čim bolj natančni.

- [5] (a) pot oz. linearen seznam z n vozlišči (P_n)
- [5] (b) cikel z n vozlišči (C_n)
- [5] (c) drevo z n vozlišči stopnje največ d
- [5] (d) poln graf z n vozlišči (K_n)
- [5] (e) mrežni graf, ki predstavlja pravokotno mrežo z v vrsticami in s stolpci, kjer je vsako izmed $v \times s$ vozlišč v mreži povezano s svojima sosedoma v isti vrstici in stolpcu (z izjemo tistih na robu mreže); vse povezave imajo enako utež

3. Obravnavali smo problem 0-1 nahrbtnika, kjer je bilo podanih n elementov z majhnimi celoštevilskimi težami t_i in vrednostmi v_i , ki smo ga reševali z dinamičnim programiranjem. Kako bi na podoben način reševali naslednje različice problema nahrbtnika? Definirajte in opišite rekurenčne relacije ter pojasnite, kako bi jih uporabili za reševanje z dinamičnim programiranjem.

- [6] (a) *Omejen nahrbtnik (bounded knapsack)*. V problemu 0-1 nahrbtnika smo lahko vsak element vzeli ali ne. Tu pa imamo na voljo n tipov elementov in za vsak tip imamo na voljo c elementov tega tipa, ki jih lahko vzamemo poljubno količino (če seveda ne presega c).
- [7] (b) *Neomejen nahrbtnik (unbounded knapsack)*. Problem je podoben omejenemu nahrbtniku, vendar imamo na voljo neomejeno zalogo vsakega izmed n tipov elementov ($c = \infty$).
- [6] (c) *Večdimenzionalen 0-1 nahrbtnik (multi-dimensional 0-1 knapsack)*. Poleg vrednosti in tež elementov imamo podane še njihove glasnosti g_i . Pri izbiri vsebine nahrbtnika moramo paziti, da tudi vsota glasnosti ne preseže podane tolerance G , ki bi pritegnila pozornost.
- [6] (d) *Problem razbitja (partition problem)*. Podan imamo seznam n števil x_i . Problem razbitja sprašuje, ali obstaja razbitje na dve skupini, ki imata enako vsoto (vsako število mora biti del ene ali druge skupine). Kako bi prevedli problem razbitja na problem 0-1 nahrbtnika? Povedano drugače, kako bi implementirali funkcijo `razbitje(stevila)` z uporabo funkcije `nahrbtnik(teze, vrednosti, nosilnost)` in zakaj?

4. K -tiška kopica je posplošitev dvojiške kopice ($k = 2$) z uporabo poravnane drevesa z vozlišči stopnje k , kjer je k parameter podatkovne strukture (in ne nujno konstanta).

- [7] (a) Opišite postopek vstavljanja elementa v k -tiško kopico in brisanja najmanjšega elementa iz nje. Indeksov staršev in otrok vam ni treba definirati.
- [8] (b) Navedite in utemeljite časovni zahtevnosti operacij vstavljanja in brisanja najmanjšega elementa.
- [10] (c) Na izbiro imamo k -tiško kopico s $k = 2$ ali $k = \sqrt{n}$. Izvedli bomo x operacij na kopici z n elementi. Od tega je b operacij brisanja najmanjšega elementa in v operacij vstavljanja elementa ($x = b + v$). Pri kakšnem deležu (asimptotično) operacij brisanja (b/x) je smiselno uporabiti katero izmed kopic?