

Podatkovne strukture in algoritmi
(2018/19)
3. domaca naloga

1. naloga: Razpršena tabela velikosti m . Spomnimo, da ima dobra funkcija zgoščanja lastnost, da ima vsak ključ k enako verjetnost $\frac{1}{m}$, da se preslika na katerokoli mesto v tabeli.

- (i) Predpostavimo, da imamo dobro funkcijo zgoščanja $h: U \rightarrow \{0, 1, \dots, m-1\}$. Kakšna je verjetnost, da se trije paroma različni elementi $u_1, u_2, u_3 \in U$ s funkcijo h preslikajo na isto mesto v tabeli (to je, $h(u_1) = h(u_2) = h(u_3)$)?
- (ii) Predpostavimo, da imamo dobro funkcijo zgoščanja $h: U \rightarrow \{0, 1, \dots, m-1\}$ in da sovpadanje rešujemo z veriženjem. Recimo, da vstavimo tri elemente v prazno razpršilno tabelo T . Kakšna je verjetnost, da sta $T[0]$ in $T[1]$ prazna?
- (iii) Naj bo $|U| = m^2$ in predpostavimo, da sovpadanje rešujemo z veriženjem. Pokażite, da za **vsako** funkcijo zgoščanja $h: U \rightarrow \{0, 1, \dots, m-1\}$ obstaja zaporedje m vstavljanj, ki porodi povezan seznam dolžine m .
Pozor: Trditev je potrebno pokazati za vsako funkcijo zgoščanja, tudi za tiste, ki niso dobre.

i)
Verjetno
st, da se
trije
paroma
različni
elementi
 $u_1, u_2,$
 u_3
pripadaj
o U s
funkcijo
 h
preslikaj
o na isto
mesto v

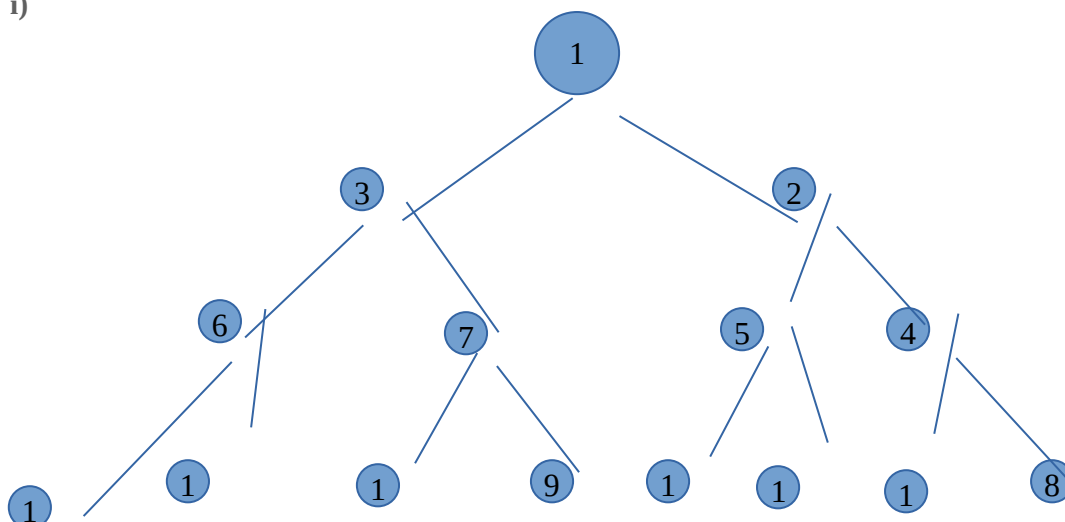
tabeli je $(1/m)^3$. Trikrat ponovimo naso sekvenco, ki je $1/m$, yato imamo, potem vrednost na kubik.

ii) Verjetnost, da sta $T[0]$ in $T[1]$ prazna je $(1-2 \times 1/m)^3$. Imamo dva elementa v tabeli $T[0]$ in $T[1]$ to nam predstavlja 2 v enacbi, ker morata biti prazna in enaka sta oba 100%. Kubik je pa zato ker vstavljamo 3 elemente v tabelo.

2. naloga: Binarno iskalno drevo in kopica.

- (i) V prazno kopico vstavimo naslednje elemente 10, 12, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13, in 2. Narišite sliko končne strukture.
- (ii) Napišite algoritem, ki v času $\mathcal{O}(n)$ preoblikuje binarno iskalno drevo T z n elementi v kopico, ki vsebuje enake elemente kot T .
- (iii) Ali obstaja algoritem, ki v času $\mathcal{O}(n)$ preoblikuje kopico K z n elementi v binarno iskalno drevo, ki vsebuje enake elemente kot K ? Odgovor utemeljite.

i)



```

vKopico(){
if(leftChild==null && rightChild==null)
kopica.insert(this.key);
return;
else if(leftChild==null){
kopica.insert(this.key);
rightChild.vKopico();
return;
else if(rightChild==null){
leftChild.vKopico ();
kopica.insert(this.key);
else{
left.vKopico ();
kopica.insert(this.key);
right.vKopico ();
return;
}
}
}

```

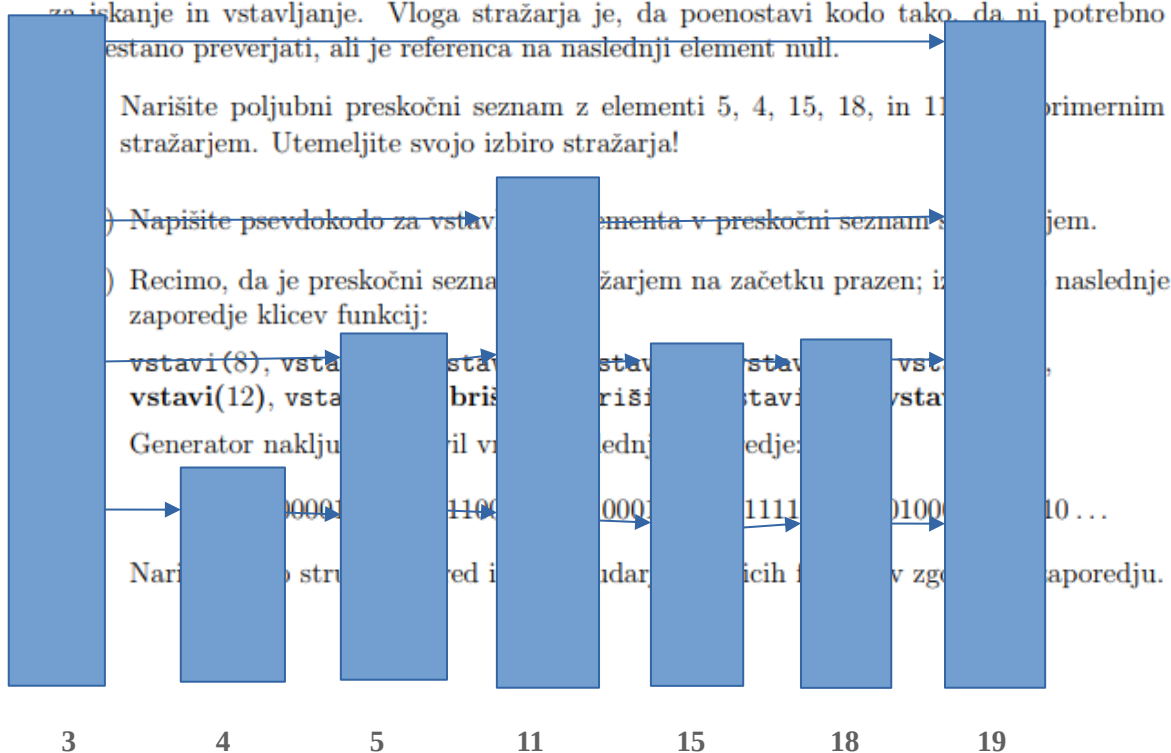
Obstaja takšen algoritem, če bi bila kopica označena od najmanjšega v korenu navzdol. Binarno drevo tako ne bo uravnoveseno, saj bomo vstavljali vse v desno. To pomeni, da bomo vstavljali proti koncu binarnega drevesa kar je $O(n)$. Za n elementov sklepam, da moramo narediti n korakov, zato je čas $O(n^2)$.

I)

Napišite pseudokodo za vstavitev elementa v preskočni seznam. 10.000

```
vstavi(8), vstavi(9), vstavi(10), vstavi(11), vstavi(12), vstavi(13), vstavi(14), vstavi(15),  
vstavi(16), vstavi(17), vstavi(18), vstavi(19), vstavi(20), vstavi(21), vstavi(22), vstavi(23),
```

Naravno, po strukturi red inendarstva, ki jih formirajo v zgoščenem zaporedju.



Izbral sem stražarja 3 in 19 zaradi tega, da sem umejl naš seznam saj 3 je najmanjša številka in vse kar vstavljamo je večje. 19 Pa zaradi tega, ker je največje število v seznamu in ne vstavljamo večjega elementa kot 19.