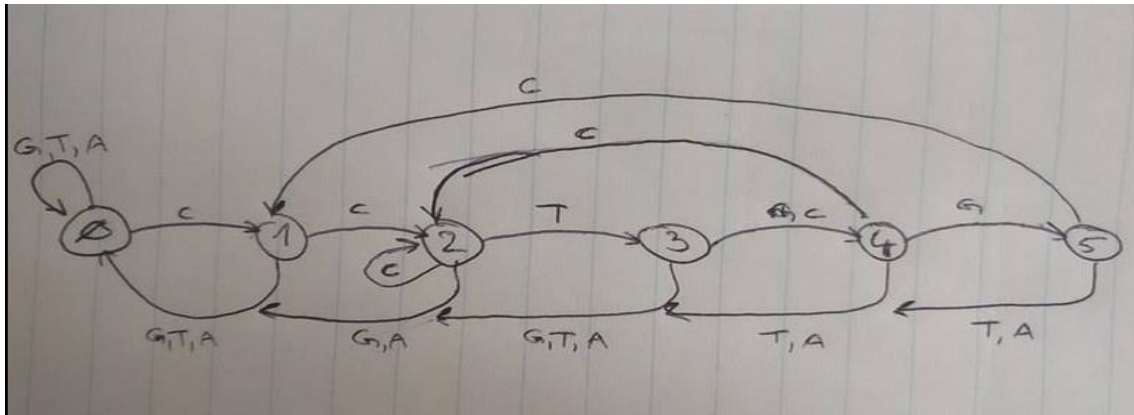


Naloga 4.1

a.)

i)



STATE	C	G	T	A
0	1	0	0	0
1	2	0	0	0
2	2	0	3	0
3	4	0	0	0
4	2	5	0	0
5	1	0	0	0

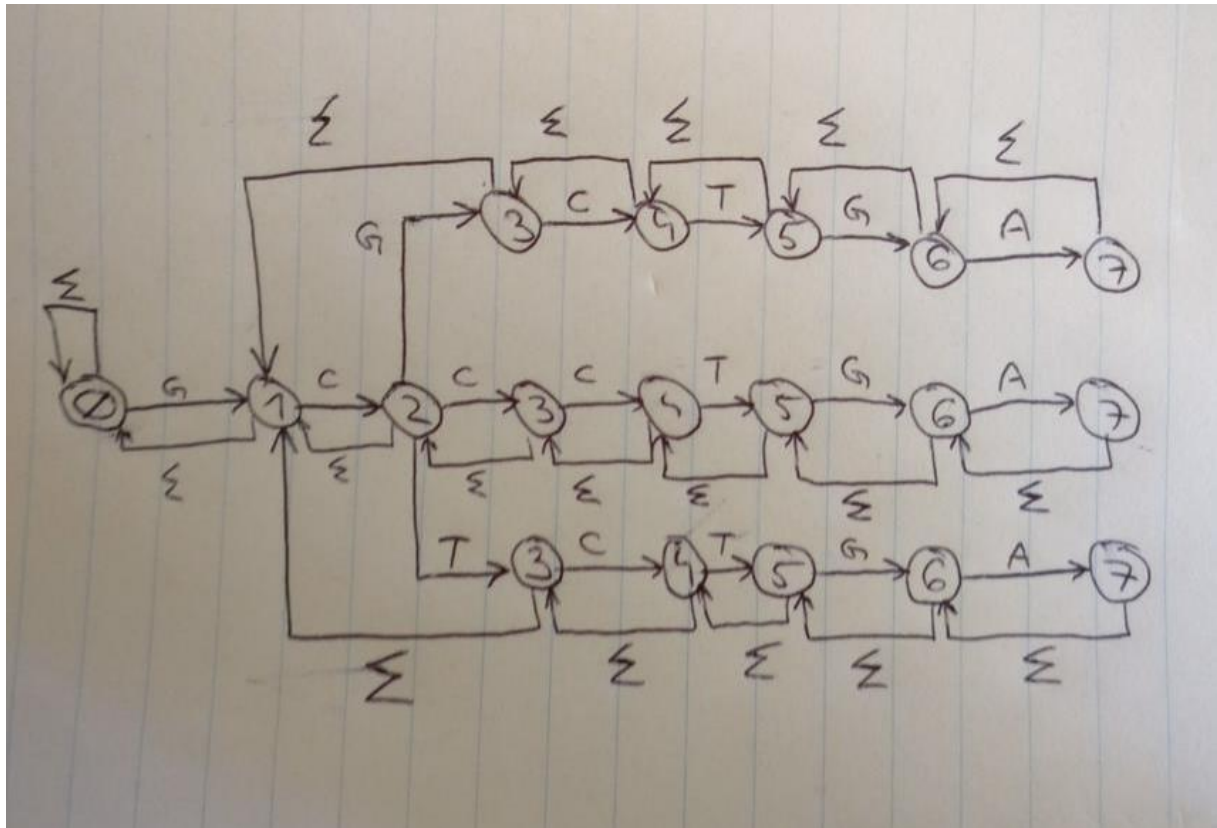
ii)

c	g	t	c	a	g	c	t	c	g	c	c	t	c	c	t	c	g	c	t	c	c	t	c	g	a
1	0	0	1	0	0	1	0	1	0	1	2	3	4	2	3	4	5	1	0	1	2	3	4	5	0

iii)

v primeru da najdemo niz se avtomat premakne če je C gremo v 2. Stanje če je pa T ali A pa v 0 stanje.

b.)



Naloga 4.2

- a) *Opišite, kako boste uporabili številsko drevo za shranjevanje vnosov v usmerjevalni tabeli in opišite, kaj naredi ukaz `route add` v vašem številskem drevesu*

V drevesu so predpone vseh nalogov omrežji. Paketi prihajajo na nas usmerjevalnik. Vsaka povezava pa označuje nadaljevanje vhodnega niza katerega tolmačimo. Ko nam zmanjka povezav ali pa pridemo do lista brez naslednikov dobimo tudi naslov na katerega se paketa posreduje. `Route add`, pretvori naslov v dvojiški zapis in preverja do kje se predpona naslova ujema z povezavami v drevesu ko najde mesto kjer predpona naslova ki ga dodajamo ni definirana doda novo povezavo in nov list. Če že obstaja povezava z enako prepono razdelimo vozlišče na dve povezavi, vsaka za eno predpono.

- b.) *Narisi drevo:*

- c.) *Ali so vsa vozlišča potrebna*

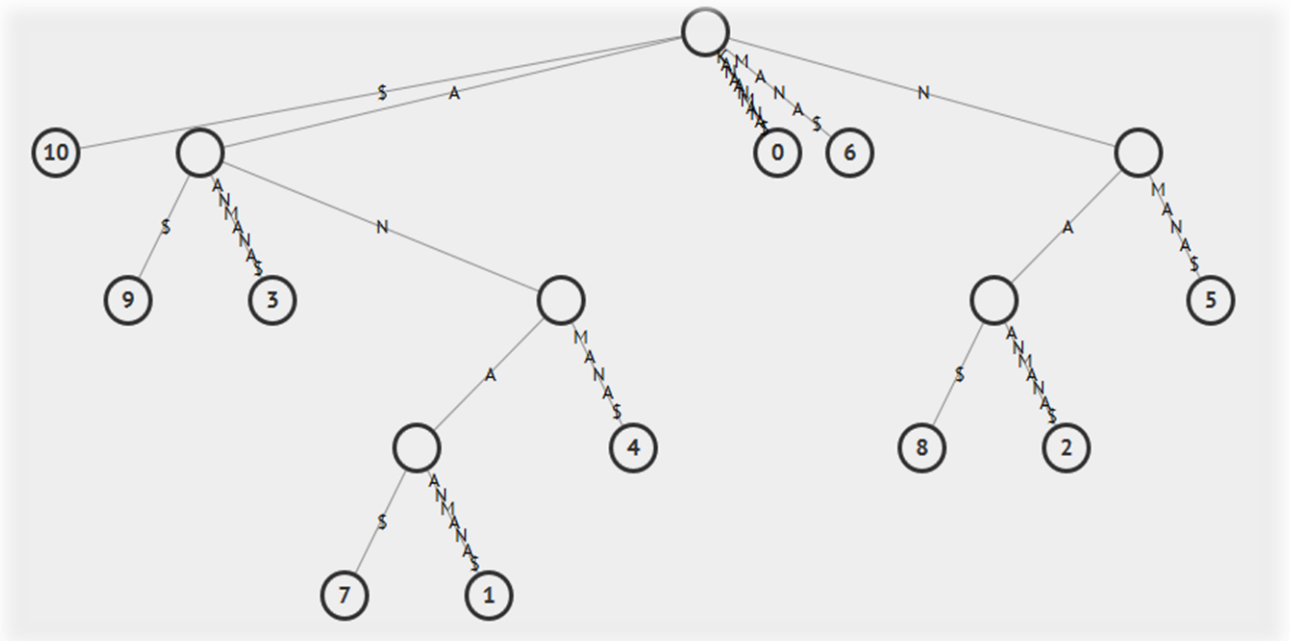
Za to drevo so vsa vozlišča potrebna saj že nazačetku pazimo da imamo točno toliko povezav kot je različnih predpon naslovov v usmerjevalni tabeli. Ker imamo shranjene samo predpone sprehod po povezavah deluje kot sekvenca znakov ki sestavljajo daljši niz. Imamo najmanjso možno porabo prostora za hranjenje usmerjevalne tabele

- d.) *Opišite, kako deluje metoda `nexthop`.*

Primerja dvojiški zapis ciljnega naslova paketa z predponami ki jih ima v drevesu. Ko preveri celotni niz oz. do zadnjega možnega lista, nam izpise ip naslov (kamor preusmerjamo) ki se nahaja na tem mestu.

4.3 naloga

a.)



Pri 0 <- je beseda KANAANMANA

c.)

Največji ponvaljaljoči se podniz je AN ponovi se 3x. KANAANMANA

d.)

V drevesu pogledamo kje je najdljasa pot do lista -> tako najdemo najdaljše ujemanje. Ker pa moramo zato iti čez celotno drevo je asimptotična zahtevnost zaradi tega $O(n^2)$. Ni ravno prijazno je pa efektivno.

e.)

Drevesi sta zgrajeni isto le da ima 2. Drevo v korenu znak c

f.)

Drevesu ST1 samo pripnemo novo vejo s celotno besedo cT