

Dovoljen je 1 A4 list z lastnimi zapiski. Druga literatura (npr. prosojnice, knjige) in elektronski pripomočki niso dovoljeni.

Nalogo rešujte v za to predviden prostor. Podpišite se na vse liste, ki jih oddate.

Na vprašanja odgovarjajte kratko (največ 2 povedi), daljši odgovori štejejo 0 točk.

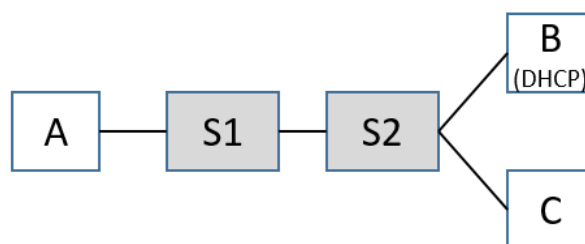
Čas pisanja je 80 minut.

izpolni ocenjevalec

1	
2	
3	
4	
SKUPAJ	

1. NALOGA (5t):

V omrežju imamo tri računalnike (A, B in C) in dve stikali (S1 in S2). Računalnik A se na novo priključi v omrežje in od računalnika B (ki je DHCP strežnik) zahteva dodelitev logičnega naslova. Med dodeljevanjem naslova poteka tudi komunikacija z računalnikom C.



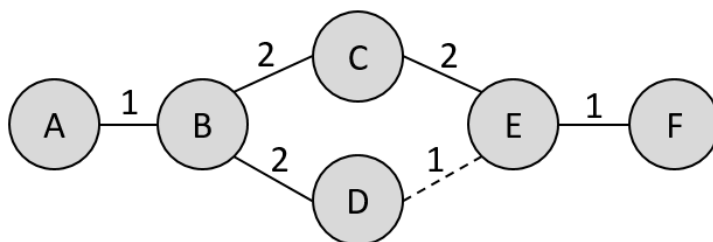
Na začetku sta obe stikalni tabeli prazni. Za spodnje zaporedje izmenjanih paketov zapiši vsebine stikalnih tabel v S1 in S2 ter akciji, ki ju izvedeta stikali. Upoštevaj, da se na povezavi namesto broadcast naslova MAC vedno uporabi unicast naslov MAC, če ga pošiljatelj le pozna (ne glede na broadcast naslavljanje pri uporabi DHCP na omrežni plasti).

	S1 stikalna tabela	S1 akcija	S2 stikalna tabela	S2 akcija
A pošlje DHCP discover				
B pošlje DHCP offer				
C pošlje okvir B				
A pošlje DHCP request				
B pošlje DHCP ack				
A pošlje okvir C				

2. NALOGA (5t):

V prikazanem omrežju usmerjevalnikov se uporablja decentralizirano usmerjanje z vektorji razdalj. Spodnja tabela prikazuje začetno stanje posredovalnih tabel, in sicer samo pravilo za najboljšo naučeno pot vsakega usmerjevalnika do usmerjevalnika F (npr. B/5 v prvem stolpcu pomeni, da A promet za usmerjevalnik F usmerja preko B po ceni 5). Po vzpostavljenem začetnem stanju tabel se v omrežju vzpostavi dodatna povezava D-E (označena s črtkano črto).

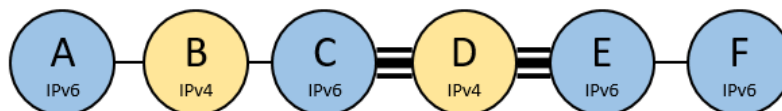
Naloga: Zapiši naslednje iteracije usmerjevalnih tabel (osredotoči se samo na usmerjanje do F), do konvergence (ustalitve usmerjevalnih tabel). Za vnos in izračune uporabi spodnjo tabelo (tabela ima odvečno število vrstic, ki jih lahko izkoristite v primeru potrebe po dodatnem prostoru).



usmerjevalnik	A	B	C	D	E	F
začetno stanje (pot do F)	B/5	A/6	E/3	B/7	F/1	F/0
vzpostavitev D-E						
1. iteracija						
2. iteracija						
3. iteracija						

3. NALOGA (5t):

V nekem omrežju imamo 6 usmerjevalnikov, kot jih prikazuje slika:



A, C, E in F uporabljajo dvojni sklad (IPv4

in IPv6), usmerjevalnika B in D pa samo IPv4. Predpostavimo, da usmerjevalniki, ki uporabljajo dvojni sklad, vedno pretvorijo paket iz IPv4 v IPv6, če ga naslednji usmerjevalnik lahko sprejme. Med usmerjevalnikoma C in E je preko usmerjevalnika D vzpostavljen tunel IPv6.

Zapiši vsebine paketov (samo relevantna polja, ki se spreminjajo: naslov pošiljatelja, naslov prejemnika, oznako toka – flow label, vrsto enkapsulirane vsebine) za naslednje primere:

a.) Usmerjevalnik A pošlje paket usmerjevalniku F. Skiciraj vsebino paketa, ki ga vidi usmerjevalnik D.

b.) Usmerjevalnik A pošlje paket usmerjevalniku F. Skiciraj vsebino paketa, ki ga prejme usmerjevalnik F.

c.) Usmerjevalnik F pošlje paket usmerjevalniku A. Skiciraj vsebino paketa, ki ga vidi usmerjevalnik D.

d.) Usmerjevalnik F pošlje paket usmerjevalniku A. Skiciraj vsebino paketa, ki ga prejme usmerjevalnik A.

4. NALOGA (5t):

Spodnji izsek prikazuje tri vrste sporočil (v obliki JSON), ki si jih izmenjujeta strežnik in klient z naše aplikacije z laboratorijskih vaj (ChatServer.java in ChatClient.java).

```
[RKchat-server] [56192] : {"format":"login", "username":"johnsnow"}
[RKchat-server] [56192] : {"datetime":"07.04.2019 10:59:44", "format":"message",
    "message":"winter is coming", "from":"johnsnow"}
[RKchat-server] [56192] : {"datetime":"07.04.2019 11:00:54", "to":"daenerys",
    "format":"private", "message":"how are you", "from":"johnsnow"}
```

Z odgovori na naslednja vprašanja poskusi v takšni implementaciji pogovorne aplikacije poiskati ekvivalente konceptov iz komunikacijskih protokolov, ki smo jih obravnavali na predavanjih.

a.) Kateri del zgornjih sporočil igra vlogo polja za multipleksiranje (kot pri IP, Ethernet, PP)?

b.) V zgornjih primerih vidimo, da strežnik ChatServer uporablja naslavljanje uporabnikov preko njihovih uporabniških imen, ki igrajo vlogo *logičnih naslovov*. Kaj je strežnikova analogija *fizičnih naslovov*, s katerimi strežnik dejansko naslovi uporabnike v postopku pošiljanja sporočil?

c.) Kaj je v vaši implementaciji strežnika bil ekvivalent nalogi, ki jo na povezavni plasti izvaja protokol ARP (upoštevajoč logične in fizične naslove iz prejšnjega vprašanja)?

d.) Denimo, da lahko strežnik hipotetično obdelava samo sporočila (vsebina polja "message") neke v naprej določene največje dolžine. Kako bi analogno z delovanjem fragmentacije pri IPv4 lahko dopolnili obliko sporočila JSON, da lahko predolgo besedilo v polju "message" razbijemo v več sporočil JSON?

Upoštevajte, da pri besedilu ne obravnavamo več posameznih bitov, ampak znake. Kot odgovor na vprašanje zapišite besedilo "Danes je lep dan, jutri pa še lepši", razbito v tri sporočila JSON. Največjo sprejemljivo dolžino besedila za eno sporočilo si izberite sami, nepomembna (konstantna) polja sporočila pa lahko izpustite.