

1 Uppgift 1

Subnät, antal bitar som krävs:

1. 63 interfaces $2^6 = 64$
2. 95 interfaces $2^7 = 128$
3. 16 interfaces $2^4 = 16$

Nätmasker:

1. $32 - 6 = 26$
2. $32 - 7 = 25$
3. $32 - 4 = 28$

Började att ta fram IP adresserna för subnät 2), då den hade som flest interfaces, där efter 1), och slutgiltigen 3). Vänstra delen nedan i IP adressen är den statiska delen, där efterföljande 'x' är möjliga IP adresser som subnätet kan använda.

IP adress:

1. 223.1.17. 10xx xxxx : 64 unika möjliga subnäts adresser, där 223.1.17.128/26 är subnätets statiska IP , då adresserna kommer vara mellan 223.1.17.128-223.1.17.191
2. 223.1.17. 0xxx xxxx : 128 unika möjliga subnäts adresser, där 223.1.17.0/25 är subnätets statiska IP, då adresserna kommer vara mellan 223.1.17.0 - 223.1.17.127
3. 223.1.17. 1100 xxxx : 16 unika möjliga subnäts adresser, där 223.1.17.192/28 är subnätets statiska IP, då adresserna kommer vara mellan 223.1.17.192-223.1.17.207

2 Uppgift 2

Ett UDP datagram har en header-längd på 16-bits (2 bytes). Dom 3 första paketen får då längden 1024 bytes och det fjärde paketet får längden 476 bytes.

3 Uppgift 3

Med 5 kollisioner väljes K-värdet utifrån $2^5 - 1$ vilket ger 31 olika värden. Det blir då en 3.2% att 4 blir K-värdet. Fördröjningen kan sedan räknas ut som $\frac{4*512}{8*100*10^6}$ vilket blir 2560 nanosekunder.

4 Uppgift 4

Alla ARP-tabeller uppdaterade:

1. A kollar upp F's MAC-adress i sin tabell efter F's IP-adress.
2. A skickar paketet med F's MAC-adress som destination.
3. Paketet routas genom nätverket till F.

A's ARP-tabell är tom:

1. A broadcastar en ARP-query med F's IP-adress vilket alla i nätverket mottager.
2. F skickar ett svar med sin MAC-adress till A.
3. A's tabell uppdateras med F's MAC-adress och det föregående scenariot utspelas.