1. Blandade frågor 4p

1a) Se avsnitt 2.2.2 i kursboken

Beständigt ”persistent” HTTP innebär att webbklienten skapar en TCP-anslutning för att hämta HTML-filen och servern behåller denna anslutning öppen (Connection: Keep-Alive) tills klienten har hämtat de tillhörande objekten. Detta innebär att klienten slipper skapa en TCP-anslutning för att hämta varje objekt. På detta sätt minskar man den totala tiden och de allokerade resurserna.

(i) Utan ”pipeling” seriell hämtning hämtar klienten klart HTML-filen och sedan begär objekten en efter en dvs. klienten vänter på svar om ett objekt innan den begär nästa. Detta medför att en tid på minst 2*RTT behövs att skapa TCP-anslutningen och hämta bas-filen och sedan en tid RTT för varje objekt; totalt (2+ **n**)*RTT. I detta fall **n** = 4 och tiden blir minst **6*RTT**.

(ii) Med ”pipeling” parallell hämtning hämtar klienten klart HTML-filen och sedan begär objekten ”back-to-back” dvs. klienten behöver **inte** vänta på svar om ett objekt innan de begär nästa. Klienten tillåts att begära fler objekt samtidigt oberoende av varandra för att hämta de **n** tillhörande objekten på webbsidan och på detta sätt snabba på svarstiden från servern; totalt **3*RTT**.

*För motsvarande rum-tids diagram hänvisas till **Figure 2.7** i kursboken och **slide 21-22** i Kapitel_2 föreläsningsbilder.* *(2p)*

1b) Se avsnitt 7.3.1 & 7.3.2 i kursboken

Paket-jitter är variationen i fördröjningen som drabbar paket över Internet, så att paket tillhörande samma multimediaströmmen kommer till mediaspelaren hos mottagare olika fördröjda trots att de skickades i konstant takt från sändaren. Det är fördröjning pga att kötiden i routrarna på vägen mellan sändaren och mottagren, varierar mest från paket till paket. Routrarnas köar bildas olika långa efter trafikintensitet som varierar över tiden.

Applikationen levererar mediaströmmen inkapslat i IP-paket som kan komma fram till mediaspelaren helt olika fördröjda. En omedelbar uppspelning skulle resultera en ojämn media (eller t.ex hack i ljud). Därför fördröjs uppspelningen av strömmande multimedia hos mottagaren något lämplig tid för att motverka jitter genom att mediaspelaren buffrar tillräckligt många paket som anländer i ojämn takt och sedan använder buffrade paket för att spela upp media i jämn takt. *(2p)*

2. Transportprotokollen 8p

2a) Se avsnitt 3.1.2 i kursboken

Den ena typen TCP är en tillförlitlig, förbindelse-orienterad dataöverföring av sekvens-numrerad byte-ström med många kontroll-mekanismer för bl.a. iordningsleverans, flödesreglering och stockningskontroll. Denna transporttjänst används för applikationer som inte tolerar paketförlust eller datafel.

Den andra typen UDP är en förbindelselös överföring av applikationsdata i form av datagram utan någon nämnvärd kontroll. Denna transporttjänst används för applikationer som kräver snabbhet eller är tidskänsliga.

Applikationsprotokoll som HTTP, FTP, SMTP, TELNET kräver felfri och kontrollerad transport av användardata och därför använder TCP.

(2p)

Observera att svaren redovisas kortfattat och hänvisas till kursboken för utförligare svar.

2b) Se avsnitt 3.3 i kursboken

DNS använder främst User Datagram Protocol (UDP) på port nummer 53 för att transportera förfrågningar och svaren. DNS-meddelanden består av en enda DNS-förfråga från klienten följt av en enda DNS-svar från servern. Dessutom är DNS-meddelandet kompakt och kort dvs innehåller data på ett antal fördefinerade fält som vanligen inte överstiger 512 byte. Att använda UDP är snabbare och att det kommer att vara minimal overhead för att få svar. På så sätt kan servern hantera enkelt fler förfrågor utan att behöva hantera onödvändiga TCP-anslutningar.

(2p)

2c) Se avsnitt 3.5.5 i kursboken

Flödeskontroll syftar till att anpassa sändaren till mottagarens förmåga att ta emot data och att inte översvämma den dvs. att förhindra TCP-sändaren from att överbelasta **mottagaren** med data som mottagaren inte hinner med (och inte har tillräcklig plats i bufferten).

Genom att mottagaren informerar sändaren kontinuerligt om hur mycket plats finns kvar i mottagarbufferten. Denna information kallas ”receive window” och finns i ett 16-bit fält i headern på varje TCP-segment, t.ex. ACK-segment som skickas till sändaren. Värdet anger hur många byte mottagaren är beredd att ta emot för tillfället.

(2p)

2d) Se avsnitt 3.7 i kursboken

Syftet med stockningskontroll är att förhindra TCP-sändaren from att överbelasta nätverket dvs. **routrarna** med paket som dem inte hinner med att vidarebefordra vilket leder till långa köer och evenuellt paketförlust.

Det är att TCP-protokollet på varje sändare (varje TCP/IP-värd) som kontrollerar stockningen på end-to-end basis. Det är kollektivt ansvar för alla Internetsanslutna TCP/IP värdar, till skillnad från en nätverksbaserad stockningskontroll.

Genom att varje TCP-sändare håller en variabel kallas för ”Congestion Window” CongWin som anger hur mycket data (t.ex. i antal segment) som sändaren får skicka i väg på en gång utan att behöva vänta på ACK på varje segment.

”Slow Start” är det då sändar-TCP börjar försiktigt genom att sända bara ett segment först och sedan ökar sändingshastighet genom att fördubbla antalet segment i CongWin efter varje RTT dvs. exponentiell ökning (om det får ACK på alla tidigare sända segmenten) tills det når en tröskel. Vid början av sändningen sätts tröskelen till ett default värde.

”Congestion Avoidance” är det då sändar-TCP har nått tröskeln och börjar öka CongWin med ett segment i taget efter varje RTT dvs. linjär ökning (om det får ACK på alla tidigare sända segmenten) för att undvika stockningen.

Händer det timeout när sändar-TCP väntar på ett ACK återgår TCP i alla fall till ”Slow Start” med ett nytt tröskelvärde, men om ett trippel duplikat ACK tas emot för ett och samma segment, indikerar detta en kortvarig stockning. Detta innebär att ett tidigare segment bland de sända segmenten är förlorat. TCP övergår till ”Congestion Avoidance” efter ”Fast Recovery” dvs. sända om segmentet. Nya värdet på tröskeln anger storleken i antal segment på halva “Congestion Window” vid senaste händelse.

(2p)

Observera att svaren redovisas kortfattat och hänvisas till kursboken för utförligare svar.

3. Ethernet & Trådlöst LAN 8p

3a) Se avsnitt 5.4.1 i kursboken

Först kontrollerar datorn om mottagarens IP-adress är på samma subnät som den själv eller inte.

	värddatorn	servern	
IP-adress	129.016.212.119	129.016.211.118	
Subnetmask	255.255.252.0 00	255.255.252.000	
----------	----------	----------	AND
Ej samma subnät	129.016.**212**.0 00	129.016.**208**.000	

Efter kontrollen, kan datorn inte kommunicera direkt med servern (på annat subnät) och därför söker sig till subnätets router (standard-gateway) för leverans av paketen. Eftersom paketen måste kapslas in i Ethernets MAC-ramar, behöver datorn ta reda på routerns MAC-adress.

ARP-tabellen var tom och inga poster hittades och datorn använder ARP-protokollet för att fråga den som har IP-adressen 129.16.213.23 att den skall svara med sin MAC-adress.

När datorn får ARP-svar från routern om MAC-adressen, sparar det i ARP-tabellen och börja sända paket adresserade till serverns IP-adress (129.16.211.118) men inkapslas i ramar adresserade till routerns MAC-adress. *(2p)*

3b) Se avsnitt 6.3.2 i kursboken

I WLAN löser man problemet genom att undvika kollisioner snarare än att försöka upptäcka dem. CSMA/CA mekanismer tillämpas kollektivt så att en trådlös STA som vill sända en ram med normal storlek, skall lyssna på radiokanalen och se att det är ledig. Är kanalen ledig, vänta DIFS, sända hela ramen och vänta på ACK (skickas av AP efter kortare väntetid SIFS).

Är kanalen upptagen, backar STA och startar nedräkning en slumpmässigt vald tid när kanalen blir ledig innan den försöker igen att sända ramen.

(2p)

3c) Se avsnitt 5.4.3 i kursboken

Switchen är en flerport brygga som kontrollerar bl.a. Ethernet-ramarnas MAC-adresser innan de skickas vidare. Switchen skickar vidare en ram till den port där mottagares MAC-adress finns med i en MAC-adress-tabell som är skapad med självlärning. Switchen kontrollerar (dynamiskt) varje inkommande ram på en switchport och läser av sändares MAC-adress i ramens header för att spara den i en MAC-adress-tabell för denna port. Switchen använder denna tabell för att avgöra till vilken port skall en ram skickas vidare om mottagares MACadress finns (är lärt finnas) i portens MAC-adress-tabell. Denna funktion har fördelen i och med att ingen konfiguration behövs för att switchen utför sitt arbete, switchen börjar direkt vidarebefordra ramarna när man ansluter den på ett nätverk (plug-and-play). *(2p)*

3d) Se avsnitt 6.3 i kursboken

Accesspunkten AP är en MAC-brygga mellan den trådlösa och den trådbundna delen av LANet. AP fungerar som basstation för de trådlösa stationerna i sitt täckningsområde. Detta innebär att AP måste presentera sig regelbundet och skapar associering med stationerna. AP har för uppgift att förmedla all trafik mellan stationerna oavsett MAC-typen (802.3 eller 802.11) och omvandlar ramarna från ena sidan till den andra. AP kan ha fler funktioner som bl.a. roaming, anpassning av bredbanden och effektspar-funktioner. *(2p)*

Observera att svaren redovisas kortfattat och hänvisas till kursboken för utförligare svar.

4. IPv4: Adresser och Subnetting 8p

Se avsnitt 4.4.2 i kursboken. Se också Lab-2 och extra materialet om IP-adressering

28.6.88.0/21

28.6.92.0/22 *28.6.**88.0**/22*

28.6.**92.0**/23 *28.6.**94. 0**/23*

28.6.**92.0**, 28.6.**92.64**, 28.6.**92.128**, 28.6.**92.192**, 28.6.**93.0**/26, 28.6.**93.64**, 28.6.**93.128**, 28.6.**93.192**/26

28.6.**93.64**, 28.6.**93.96**, 28.6.**93.128**, 28.6.**93.160**, 28.6.**93.192**/27, 28.6.**93.224**/27

28.6.**93.224**, 28.6.**93.228,** 28.6.**93.232**, 28.6.**93.236**, 28.6.**93.240**/30, 28.6.**93.244**, 28.6.**93.248**, 28.6.**93.252**

4a) Tredje byte (från vänster) i adressen är 88 decimalt som är **01011**000 binärt där de första fem bitarna tillhör prefixen i adressblocket. Man använder 6e bit för att dela upp adressblocket i två lika stora delar. Tredje byte blir 88 = **01011**000 och 92 = **01011**100.

Den första halvans prefix blir 28.6.**88**.0/22 som reserveras av ISP.
Den andra (övre) halvans prefix blir 28.6.**92**.0/22 som kunden kan få sin nätverksadress ifrån. Varje del har utrymme på max 1022 hostadresser.

(2p)

4b) Eftersom antalet IP adresser för företagets nätverk skall vara ca 470, är det närmaste binära storleken på ett subnät 512 vilket innebär *halva* 28.6.**92**.0/22.

$512 = 2^9$ som betyder att prefixet skall ha 9 bitar för företagets hostadresser.
Nätverksdelen blir 32 – 9 = 23 bitar och således blir prefixet:

28.6.**92**.0/23 alternativt 28.6.**94**.0/23 och mask 255.255.254.0. I fortsättningen är det första (lika bra det andra) alternativet som används för företagets nätverk.

(2p)

4c) Tredje byte (från vänster) i adressen är 92 decimalt som är **01011**100 binärt där de första sju bitarna tillhör företagets prefix. Genom att använda sista bit i denna byte och de första två bitarna i fjärde byte får man åtta stycken stora subnät med 6 bitar för hostdelen (62 adresser). Subnäten är som följande:

28.6.92.0, 28.6.92.64**, 28.6.92.128,** 28.6.92.192/26**,**
28.6.93.0, 28.6.93.64**, 28.6.93.128,** 28.6.93.192/26
och alla har samma subnätmask 255.255.255.**192**

Fem av dessa subnät kan nu användas för företagets 5 subnät med 62 adresser vardera. De resterande subnät skall subnettas ytterligare:

28.6.93.64/26 till två mindra subnät 28.6.93.**64**/27 och 28.6.93.**96**/27
28.6.93.128/26 till två mindra subnät 28.6.93.**128**/27 och 28.6.93.**160**/27
28.6.93.192/26 till två mindra subnät 28.6.93.**192**/27 och 28.6.93.**224**/27

Fem av dessa subnät kan nu användas för företagets 5 subnät med 30 adresser vardera och alla har samma subnätmask 255.255.255.**224**. Det subnätet 28.6.93.**224**/27skall subnettas ytterligare till 8 stycken 30-bitars subnät:

28.6.93.**224, 228, 232, 236, 240, 244, 248, 252**/30 och subnetmask 255.255.255.**252**

Fem av dessa subnät kan nu användas för företagets 5 subnät (point-to-point länkar) med 2 hostadresser vardera. De resterande subnäten 28.6.93.**244, 248, 252**/30 förblir oanvända. *(2p)*

Observera att svaren redovisas kortfattat och hänvisas till kursboken för utförligare svar.

5. Routing 6p

5a) *(2p)*

- **”distance vector”** om, via noden, **nåbar destination**, **vägkostnaden** och next hop.
- **”link state”** om nodens identitet, grannroutrar, status och kostnad av alla aktiva länkar som noden är direktansluten till.

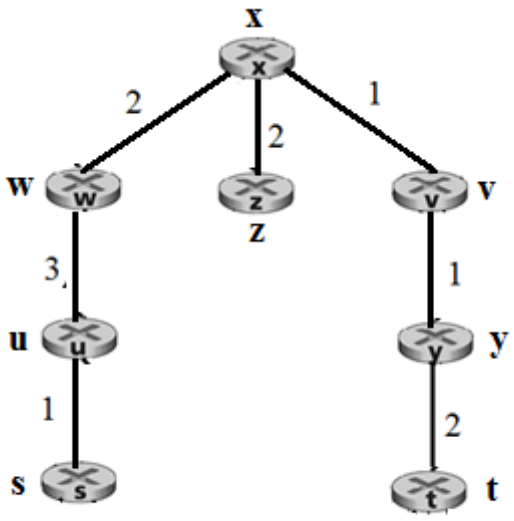
5b) Dijkstra’s algoritm *(2p)*

Steg	N’	D(s),p(s)	D(t),p(t)	D(u),p(u)	D(v),p(v)	D(w),p(w)	D(y),p(y)	D(z),p(z)
0	x	∞	∞	∞	1,x	2,x	∞	2,x
1	xv	∞	∞	∞	----	2,x	2,v	2,x
2	xvw	∞	∞	5,w		----	2,v	2,x
3	xvwz	∞	7,z	5,w			2,v	----
4	xvwzy	∞	4,y	5,w			----	
5	xvwzyt	7,t	----	5,w				
6	xvwzytu	6,u		----				
7	xvwzytus	----						

5c) Routing-tabell för nod ”x” *(1p)*

Destination	Next hop	Cost
s	w	6
t	v	4
u	w	5
v	x	1
w	x	2
y	v	2
z	x	2

5d) Graf: nod ”x” **är root** *(1p)*



Observera att svaren redovisas kortfattat och hänvisas till kursboken för utförligare svar.

6. nslookup 6p

6a) Studenten frågade efter namnen på email-servrar för domänen **ericsson.com**. Studenten specificierat **RR**-typen med **-type=MX**, som står för ”Mail eXchanger” och efter domännamnet **ericsson.com** angav studeten vilken namnserver ”**dns.uu.se”** skall kontaktas av DNS-klienten. *(1p)*

6b) DNS-förfrågan skickades till DNS-servern som har hostnamn **dns.uu.se** och IP-adress 130.238.7.10. Datorns DNS-klient har fått ta reda på IP-adressen till denna server genom att skicka DNS-förfrågan till en av Chalmers lokala DNS-servrarna.

DNS Servers . . . . . : 129.16.1.53
129.16.2.53

(1p)

6c) Det står ” Non-authoritative answer” därför att svaret kom från **dns.uu.se** vilken är inte officiell ”authoritative” DNS-server för domänen **ericsson.com**. Denna server erdjuder rekursion och kontaktar andra namnservrar för att skaffa DNS-informationen. När Studentens DNS-klient skickade **nslookup**-förfråga till **dns.uu.se**, skickade denna server en fråga till Root/TLD DNS-servrar för att ta reda på de officiella ”authoritative” DNS-servrar för domänen **ericsson.com**.

ericsson.com nameserver = ns2.ericsson.se
ericsson.com nameserver = ns1.ericsson.se
ericsson.com nameserver = e3dns.ericy.com

Sedan skickade denna server en fråga till en av Ericssons namnservrar för att få svar om domänens email-servrar. *(2p)*

6d) De olika delarna av DNS-information som svaret innehåller, är:

- Själva svaret, dvs. namnen på Ericssons email-servrar med deras preferens
- Namnen på domänens officiella ”authoritative” namnservrar
- Extra information om IP-adresser för namnservrarna

(2p)