

Naam	Initialen	Studentnummer	Studierichting

**Tentamen Telematica Systemen en Toepassingen (261000)**  
**5 november 2008, 9.00 – 12.30**

Opmerkingen:

- Alleen 1 dubbelzijdig blad met aantekeningen / samenvatting (ongeacht lettergrootte / dichtheid) en een woordenboek zijn toegestaan als hulpmateriaal. Het gebruik van het boek van Kurose en Ross of enig ander gedrukt materiaal is niet toegestaan.
- Het gebruik van een rekenmachine is toegestaan. Gebruik van PDA, laptop computer, mobiele telefoon, enz., is niet toegestaan. Schakel je mobiele telefoon uit en berg hem op.
- Geef je antwoorden op deze bladen.
- Aanduidingen zoals “[10]” bij vragen betekenen dat je 10 punten voor die vraag kunt verdienen.
- Vul je naam, studentnummer, enz., bovenaan deze bladzijde in.

Alleen voor de docent:

Vraag	1	2	3	4	5	6	Totaal
Punten							
Maximum	5	14	12	18	11	8	68

**Lijst van afkortingen:**

ACK	ACKnowledgement
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IP	Internet Protocol
MAC	Medium Access Control
NAT	Network Address Translator
NS	Name Server
PDU	Protocol Data Unit
RTT	Round Trip Time
TCP	Transmission Control Protocol
TLD	Top Level Domain
URL	Universal Resource Locator
VC	Virtual Circuit

**1) Algemene vragen [5]**

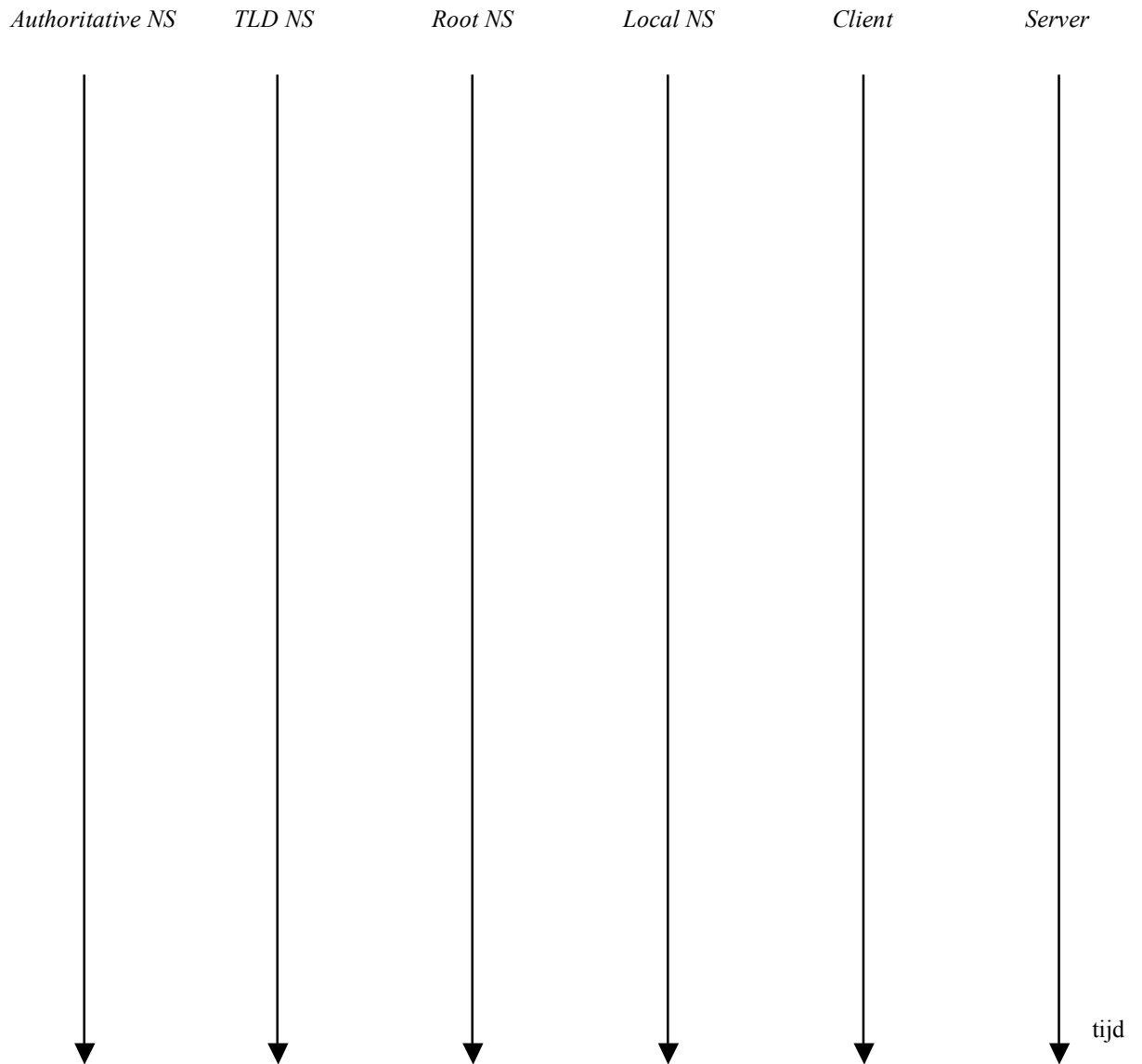
Geef van ieder van de volgende stellingen aan of deze waar of niet waar is. Omcirkel het juiste antwoord.

**Let goed op:** *goed antwoord: +1/2 punt; fout antwoord: -1/2 punt; geen antwoord: 0 punt.*

- a) Hoewel niet gebruikelijk, kunnen MAC adressen wel via DHCP worden toegewezen.  
waar / niet waar
- b) *Circuit switching* heeft als voordeel, ten opzichte van *packet-switching*, dat service-kwaliteit beter kan worden gegarandeerd.  
waar / niet waar
- c) Een lagere *gesampelde* RTT kan in TCP leiden tot een grotere waarde voor het gekozen *time-out* interval.  
waar / niet waar
- d) Het gebruik van cumulatieve ACKs heeft als nadeel dat de ACK pakketten groter worden.  
waar / niet waar
- e) Het veelvuldig gebruik van NAT versterkt de problematiek rondom de te kleine adresruimte van IPv4.  
waar / niet waar
- f) Het type-veld in het Ethernet frame speelt een vergelijkbare rol als het protocol-veld in IP.  
waar / niet waar
- g) *Switch poisoning* (in Ethernet switches op data link laag niveau) is het gevolg van het ontbreken van effectieve maatregelen tegen *poisoned reverse* (op netwerk laag niveau).  
waar / niet waar
- h) *Go-back-N* en *Selective Repeat* zijn verschillende *reliable data transfer* mechanismen, behalve voor  $N=1$ , dan zijn ze hetzelfde.  
waar / niet waar
- i) Het TCP protocol verhoogt zijn data rate tijdens de *slow-start* fase exponentieel.  
waar / niet waar
- j) Het feit dat de transmissietijd van een ACK pakket kleiner is dan van een data pakket wordt veroorzaakt door de hogere propagatiesnelheid van kleine pakketten.  
waar / niet waar



- b) Van een bekende krijg je per SMS de URL van een nieuwe website doorgegeven, te weten [www.datwilikook.nl](http://www.datwilikook.nl). In de SMS lees je verder dat er op die site naast wat (HTML) tekst ook een mooie afbeelding te zien is. Je typt bovenstaande URL in het URL-veld van je browser in. Geef in onderstaand diagram aan welke HTTP- en DNS-berichten (dus **geen** TCP berichten) worden uitgewisseld alvorens je de nieuwe website inclusief afbeelding op je scherm ziet. De DNS servers werken daarbij iteratief en alle DNS caches zijn intitieel leeg. [4]





### 3) *Reliable Data Transfer* [12]

In deze opgave bekijken we *reliable data transfer*. In het algemeen gebruikt *reliable data transfer* hertransmissies om pakketten waarvan niet zeker is dat ze goed bij de ontvanger zijn aangekomen opnieuw te versturen.

We beschouwen eerst het *alternating bit protocol* (in het boek beschreven als rdt3.0). Neem aan dat de onderliggende service pakketten verloren kan laten gaan, of fouten kan introduceren. De onderliggende service zal **niet** de volgorde van pakketten veranderen.

Geval 1: Stel, een datapakket dat door de zender naar de ontvanger wordt gestuurd gaat verloren.

a) Hoe zal de zender dit ontdekken? [1]

Geval 2: Stel, een ACK-pakket dat door de ontvanger naar de zender wordt gestuurd gaat verloren.

b) Hoe zal de zender dit ontdekken? [1]

c) Kan de zender het verschil ontdekken tussen geval 1 en geval 2? Indien ja, hoe? Indien nee, waarom niet? [1]

Stel dat het volgende pakket dat door de zender verstuurd wordt (de hertransmissie) correct ontvangen wordt.

d) Kan de **ontvanger** het verschil ontdekken tussen geval 1 en geval 2? Indien ja, hoe? Indien nee, waarom niet? [1]

- e) Stel dat de **ontvanger** zojuist een pakket met *sequence* nummer 1 heeft ontvangen, en opnieuw een pakket met *sequence* nummer 1 ontvangt. Hoe kan de ontvanger weten of dit pakket een duplicaat is, en niet een nieuw pakket met *sequence* nummer 1, bijvoorbeeld omdat de zender tussendoor een pakket met *sequence* nummer 0 heeft gestuurd dat verloren is gegaan? [2]

Hieronder beschouwen we *pipelined* protocollen. We nemen nog steeds aan dat de onderliggende service alleen fouten en pakketverlies introduceert, en dat volgorde van pakketten behouden blijft.

- f) Beschouw het geval van *Selective Repeat*, met een windowgrootte van  $w$ . Hoe groot moet de *sequencenumber* ruimte minimaal zijn zodat de ontvanger altijd weet of een ontvangen pakket een hertransmissie of een nieuw pakket is? Leg je antwoord uit. [2]

*Reliable data transfer* wordt vaak zowel in de linklaag als in de transportlaag geïmplementeerd.

- g) Wanneer het gaat om het gebruik van de transmissiecapaciteit van het communicatienetwerk, wat is efficiënter, het doen van hertransmissies in de linklaag of in de transportlaag? Waarom? [2]
- h) Stel dat we in een netwerk *reliable data transfer* geboden wordt door de linklaag. Waarom is het dan nog nodig *reliable data transfer* te implementeren in de transportlaag? Hoe kan data nog verloren gaan, of fouten bevatten wanneer de linklaag in het gehele netwerk perfect betrouwbaar is? [2]



4) Addressering [18]

We beschouwen een pakket dat onderweg is naar een bepaalde eindbestemming. Op zijn weg zal het pakket achtereenvolgens een router, een NAT en een switch tegenkomen (zie figuur). Het pakket bevat een TCP segment met een *source port number* en *destination port number*. Het segment wordt gestuurd in een IP datagram met onder andere een *source IP address* en *destination IP address*. Op zijn beurt wordt dit IP pakket verstuurd in een MAC frame, met *source MAC address* en *destination MAC address*.



Allereerst komt het pakket aan bij een **router**. De router moet een keus maken naar welke uitgaande link (interface) het pakket doorgestuurd wordt.

a) Welk(e) van de bovenstaande zes vetgedrukte adresvelden wordt / worden gebruikt door de router om de uitgaande interface te bepalen? [1]

b) Geef voor het volgende IP adres de *dotted-decimal* notatie: [1]  
 11000001 00100000 00000000 00000101

De router heeft 3 uitgaande links, genummerd van 0, 1 en 2. Stel pakketten moeten op de volgende manier geforward worden naar de uitgaande links.

Address range	Link interface
11000001 00000000 00000000 00000000 t/m 11000001 00010111 11111111 11111111	1
11000001 00011000 00000000 00000000 t/m 11000001 00011111 11111111 11111111	2
11000001 00100000 00000000 00000000 t/m 11000001 01111111 11111111 11111111	1
Default	0

c) Geef voor deze router een *forwarding* tabel met **drie** entries, gebruikmakend van *longest prefix matching*, die pakketten naar de juiste uitgaande link zal *forwarden*. Gebruik de a.b.c.d/x (*dotted-decimal*) notatie in plaats van de binaire notatie. [3]

Address	Link interface

- d) Beschrijf hoe de router aan de hand van de *forwarding* tabel van opgave c) de juiste link interface voor het IP adres van opgave b) kan bepalen. [2]

Vervolgens komt het pakket aan in een NAT. De NAT “vertaalt” adressen van de pakketten die worden *geforward*.

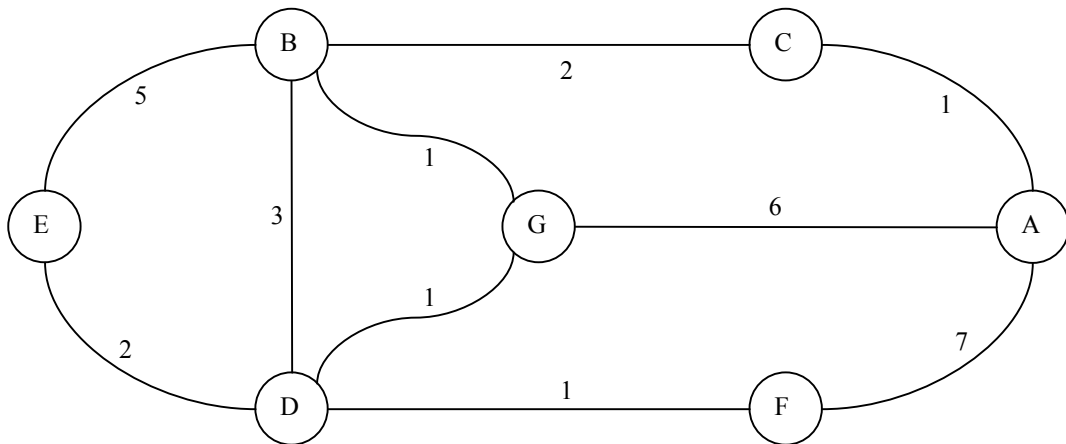
- e) Welk(e) van de zes in de inleiding van deze vraag genoemde en vetgedrukte adres- (of poortnummer) velden zal/zullen zeker gewijzigd zijn wanneer het pakket de NAT verlaat? Geef voor ieder van deze velden aan hoe het bepaald wordt door de NAT. [3]
- f) Welk(e) van de zes in de inleiding van deze vraag genoemde en vetgedrukte adres- (of poortnummer) velden zal/zullen zeker **niet** gewijzigd zijn wanneer het pakket de NAT verlaat? [1]
- g) Noem 2 verschillende manieren waarop *entries* aan de NAT tabel kunnen worden toegevoegd. Onder welke omstandigheden zal elk van de 2 manieren gebruikt worden? [2]

Nu komt het pakket aan in de **switch**. De switch heeft een switch tabel die gebruikt wordt voor het *forwarden* van pakketten.

- h) Voor ieder van de volgende drie gevallen, geef aan wat de switch met het pakket zal doen. [3]
- i. De switch tabel bevat **geen** *entry* met het *destination* MAC adres van het pakket.
  
  - ii. De switch tabel bevat een *entry* met het *destination* MAC adres van het pakket, en deze *entry* verwijst naar de link waarop het pakket binnenkwam.
  
  - iii. De switch tabel bevat een *entry* met het *destination* MAC adres van het pakket, en deze *entry* verwijst naar een andere link dan die waarop het pakket binnenkwam.
- i) Wordt het pakket ook gebruikt om een *entry* aan de tabel toe te voegen / te wijzigen? Indien ja, hoe? Indien nee, wanneer is dat wel het geval? [2]

5) Routing [11]

- a) Wat is het verschil tussen routing en *forwarding*? [1]
  
- b) Is in een *virtual-circuit* netwerk ook routing nodig? Indien nee, waarom niet? Indien ja, waarvoor? [2]
  
- c) Waarom bevat de *forwarding* tabel in een *virtual-circuit* netwerk naast *Incoming VC* en *Outgoing Interface* ook kolommen voor *Incoming Interface* en *Outgoing VC*? [2]
  
- d) Beschouw onderstaand IP-netwerk: de knooppunten zijn de routers en de lijnen de communicatie links. Gebruik het algoritme van Dijkstra om onderstaande tabel in te vullen om zodoende het kortste pad van A tot alle andere nodes te berekenen. Hierbij is  $N$  de verzameling van al afgehandelde nodes,  $D(x)$  de afstand tot node  $x$ , en  $p(x)$  de voorganger van  $x$  op het kortste pad van A naar  $x$ . [6]



step	$N$	$D(B), p(B)$	$D(C), p(C)$	$D(D), p(D)$	$D(E), p(E)$	$D(F), p(F)$	$D(G), p(G)$
0	{A}	$\infty$	1,A	$\infty$	$\infty$	7,A	6,A
1							
2							
3							
4							
5							
6							

6) **Link-layer en access mechanismen: token passing protocollen [8]**

We beschouwen een specifiek *token passing* protocol: het *token ring* systeem; in zo'n systeem mag het station dat het *token* "in bezit" heeft, een pakket (in ons geval, met vaste lengte van 16000 bits; dit is inclusief *headers*) versturen. Het *token* zelf is precies 160 bits lang. Er zijn 10 stations aangesloten. De signaalpropagatietijd is  $C = 2 \cdot 10^5$  km/s, de lengte van de kabel  $L = 100$  km, de transmissiesnelheid  $R = 8 \cdot 10^6$  bit/s. Per station treedt bovendien een vertraging op van 160 bittijden, dit geldt voor zowel pakketten als *tokens*.

a) Hoe lang is een *token*, gemeten in meters? Toon je berekening! [2]

b) Hoe lang duurt het voor een *token* om precies een keer "rond de ring te reizen" als er geen verder dataverkeer is? [2]

Wanneer een *token* bij een station aankomt, mag hoogstens één pakket verzonden worden. Direct na verzending (transmissie) van zo'n pakket, wordt het *token* weer verder gezonden. Het ontvangende station is verantwoordelijk voor het verwijderen van het pakket van de ring.

- c) We kijken nu specifiek naar één station, en gaan er van uit dat daar voldoende data klaar staat om iedere keer wanneer het token langs komt, een pakket te verzenden. Wat is in de geschetste situatie de minimale en de maximale throughput die gehaald kan worden door **dat** station.

Hint: maak twee berekeningen, één waarbij alle andere stations niets te verzenden hebben, en één waarbij alle andere stations ook telkens een packet willen versturen.

Minimale *throughput* [2]:

Maximale *throughput* [2]: