

Naam	Initialen	Studentnummer	Studierichting

Tentamen Telematica Systemen en Toepassingen (261000)

4 november 2009, 8.45 – 12.15

Opmerkingen:

- De tijdens het werkcollege ingeleverde samenvattingen, die aan het begin van het tentamen zijn teruggegeven, mogen bij het tentamen gebruikt worden. Lever deze samenvattingen na afloop van het tentamen weer in, zodat je ze bij een eventuele herkansing in januari opnieuw kunt gebruiken.
- Het gebruik van het boek van Kurose en Ross of enig ander materiaal buiten het hierboven genoemde 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	7	Totaal
Punten								
Maximum	5	16	6	14	16	13	10	80

Lijst van afkortingen:

ACK	ACKnowledgement
ARP	Address Resolution Protocol
CSMA	Carrier Sense Multiple Access
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IMAP	Internet Mail Access Protocol
IP	Internet Protocol
MAC	Medium Access Control
NAT	Network Address Translator
P2P	Peer-to-Peer
POP3	Post Office Protocol – version 3
RTT	Round Trip Time
SMTP	Simple Mail Transfer Protocol
TCP	Transmission Control Protocol
TDM	Time Division Multiplexing
TLD	Top Level Domain
TTL	Time-to-live
UDP	User Datagram Protocol
XHTML	Extensible Hypertext Markup Language

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) Een protocolbeschrijving is langer en complexer dan een servicebeschrijving.
waar / niet waar
- b) IP datagram fragmentatie is een van de oorzaken van het tekort aan IPv4 adressen.
waar / niet waar
- c) Go-Back-N is voor $N=1$ gelijk aan het Alternating-Bit Protocol
waar / niet waar
- d) Ten gevolge van *head-of-line blocking* wordt niet alle capaciteit van een router benut.
waar / niet waar
- e) Omdat het TTL veld in de IP *header* in iedere router verandert moet steeds de *header checksum* opnieuw berekend worden.
waar / niet waar
- f) Het schema voor 2-dimensionale pariteit kan alleen een fout databit corrigeren. Een fout in een pariteitsbit kan niet gecorrigeerd worden.
waar / niet waar
- g) Bij CSMA is de kans op een *collision* afhankelijk van de propagatietijd van het kanaal.
waar / niet waar
- h) Omdat switches zelflerend zijn wordt hun ARP tabel automatisch gevuld.
waar / niet waar
- i) Om het adrestekort vanwege het gebruik van IPv4 op te vangen, worden MAC adressen steeds vaker via DHCP toegewezen.
waar / niet waar
- j) Satellietcommunicatie tussen Europa en de VS leidt tot lagere vertragingstijden dan communicatie via een zeekabel.
waar / niet waar

2) Applicatieprotollen [16]

a) World Wide Web [5]

- i. Waarom maakt het HTTP protocol gebruik van TCP, en niet van UDP? [1]

- ii. Wat zijn de belangrijkste voordelen van web caches (proxy servers)? Waarom wordt er door Nederlandse universiteiten relatief weinig gebruik van gemaakt? [1]

- iii. Bij HTTP/1.0 zijn alle verbindingen non-persistent, en bij HTTP/1.1 in het algemeen persistent. Leg de verschillen tussen beide verbindingstypes uit en geef aan hoe die verschillen tot uiting komen in response tijden. [1]

- iv. Noem de belangrijkste verschillen tussen HTML en XHTML? [1]

- v. Wat zijn de belangrijkste verschillen tussen “web servers” en “web services”? [1]

b) Elektronische post [5]

- i. Waarom wordt er bij email een verschil gemaakt tussen email servers en email clients? [1]

- ii. Er zijn drie belangrijke email protocollen: SMTP, POP3 en IMAP. Zijn deze protocollen alternatieven van elkaar, of dienen ze verschillende doelen en moeten ze in combinatie gebruikt worden? Leg uit! [2]

- iii. Het SMTP email protocol schrijft voor dat alle berichten als 7-bit ascii gecodeerd moeten worden. Is het met SMTP dan wel mogelijk om plaatjes en foto's te versturen? Indien ja, hoe moet dat dan? [2]

c) Domain Name System [6]

- i. Momenteel zijn er een aantal verschillende web servers op de UT. Voorbeelden zijn <http://mycampus.utwente.nl/> en <http://blackboard.utwente.nl/>. De UT wil dat in de toekomst in principe alle UT webpagina's beginnen met <http://www.utwente.nl/>. Is het met een dergelijke structuur nog steeds mogelijk om <http://www.utwente.nl/mycampus/> en <http://www.utwente.nl/blackboard/> op verschillende web servers te draaien? Leg uit! [2]

- ii. DNS kent twee soorten queries: iterative en recursieve queries. Daarnaast kent het vier soorten DNS servers: Local, Root, Top Level Domain (TLD) en Authoritative DNS server. Welke queries worden in principe gebruikt tussen welke servers en waarom? [2]

- iii. DNS records kunnen van een verschillend type zijn. Vier veel gebruikte types zijn: Type=A, Type=NS, Type=CNAME en Type=MX. Leg het doel van deze vier types uit! [2]

3) P2P netwerk applicaties [6]

- a) Wat is, in één woord, het sleutelvoordeel van *peer-to-peer* applicaties in vergelijking met klassieke *client-server* applicaties? [1]

We beschouwen in de rest van deze opgave een typische *peer-to-peer downloading* situatie. Een bestand ter grootte F (bytes) wordt verspreid van een node (server) naar N peers. De *upload*-snelheid van de server waarop het bestand zich initieel bevindt bedraagt u_s (bytes per seconde). De *download*-snelheid van iedere *peer* (behalve de initiële server) bedraagt d_i bytes per seconde; de *upload*-snelheid van iedere peer bedraagt u_i bytes per seconde ($i=1, \dots, N$). Enige notatie: $d_{\min} = \min_i \{d_i\}$ en $\sum_i u_i = u_1 + \dots + u_N$.

Het is bekend dat de distributietijd D in het hierboven geschetste scenario minimaal gelijk is aan

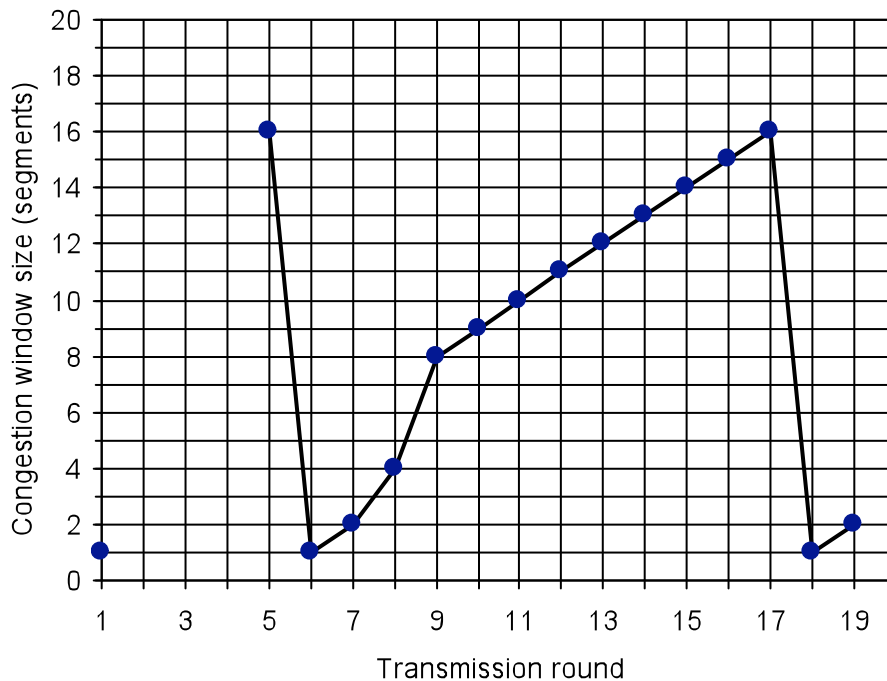
$$\min \{F/u_s, F/d_{\min}, NF/(u_s + \sum_i u_i)\}.$$

In de volgende deelopgaven vragen we je deze expressie te verklaren.

- b) Verklaar waarom $D \geq F/u_s$. [2]
- c) Verklaar waarom $D \geq F/d_{\min}$. [1]
- d) Verklaar waarom $D \geq NF/(u_s + \sum_i u_i)$. [2]

4) TCP Congestion Control / RTT estimation [14]

Beschouw het volgende verloop van de grootte van het TCP *congestion window* als functie van de tijd. De gebruikte versie van TCP is Reno (TCP met *fast retransmit* en *fast recovery*). De grootte van het *congestion window* van transmissieronde 2 t/m 4 en vanaf transmissieronde 20 is (nog) niet ingevuld in de grafiek.



- a) Gedurende welk(e) tijdsinterval(len) tussen transmissieronde 6 en 18 werkt *slow start*?
Gedurende welk(e) tijdsinterval(len) *congestion avoidance*? [2]

slow start:

congestion avoidance:

- b) Wordt na de 5^e transmissieronde pakketverlies gedetecteerd door het ontvangen van 3 *duplicate ACKs*, of door een *time-out*? En na de 17^e transmissieronde? [2]

na de 5^e transmissieronde:

na de 17^e transmissieronde:

- c) Wat zal de grootte van het *congestion window* geweest zijn gedurende de 4^e transmissieronde? [1]
- d) Wat is de waarde van de variabele *Threshold* gedurende de 6^e transmissieronde? [1]
- e) Na transmissieronde 19 vindt **geen** pakketverlies plaats. Wat is de grootte van het *congestion window* tijdens transmissieronde 20? [1]

TCP gebruikt een RTT *estimation* algoritme voor het bepalen van de lengte van het timeout interval. Zoals uitgelegd in het boek van Kurose & Ross wordt dit gedaan met behulp van de volgende 3 formules:

$$\text{DevRTT} = (1 - \beta) \cdot \text{DevRTT} + \beta \cdot |\text{SampleRTT} - \text{EstimatedRTT}|$$

$$\text{EstimatedRTT} = (1 - \alpha) \cdot \text{EstimatedRTT} + \alpha \cdot \text{SampleRTT}$$

$$\text{TimeoutInterval} = \text{EstimatedRTT} + 4 \cdot \text{DevRTT}$$

Hierbij worden de waarden $\alpha = 1/8$, en $\beta = 1/4$ gebruikt.

f) Waarom is het zo belangrijk voor TCP dat de waarde van het timeout interval goed gekozen wordt? Wat zou het gevolg zijn wanneer de waarde te groot of te klein is? [2]

g) Stel dat op zeker moment de gegeven variabelen de volgende waarden hebben; $\text{EstimatedRTT} = 80$ ms; $\text{DevRTT} = 0$ ms; $\text{TimeoutInterval} = 80$ ms. Stel dat de eerstvolgende meting van de RTT als waarde 96 ms oplevert. Wat zal dan de waarde van het timeout interval worden? [2]

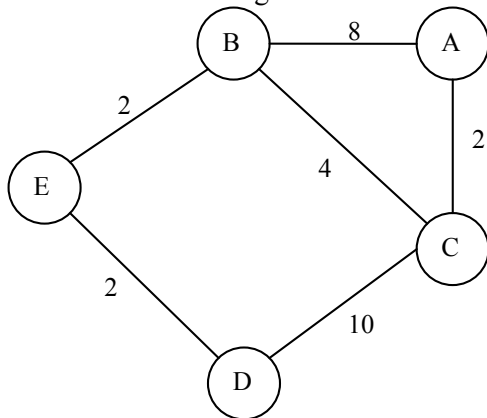
h) Stel dat (direct na hetgeen beschreven is in deelopgave g) een RTT gemeten wordt van 42 ms. Wat zal nu de waarde van het timeout interval worden? [1]

i) Hoe kan het gebeuren dat tussen twee opeenvolgende metingen de waarde van het timeout interval groter wordt, terwijl de gemeten waarde voor de RTT kleiner is? [2]

5) Routing [16]

- a) Noem 2 belangrijke functies van de netwerklaag. [2]
- b) Noem de 2 belangrijkste redenen waarom hierarchische routing gebruikt wordt (bijvoorbeeld in het Internet). [2]

Beschouw nu het volgende netwerk met de daarin aangegeven link-kosten:



- c) Gebruik Dijkstra's kortste-pad-algoritme om de kortste paden van node A naar alle andere nodes te berekenen. Gebruik het algoritme om de volgende tabel in te vullen, waarin N' de verzameling van al afgehandelde nodes is, $D(x)$ de afstand tot node x , en $p(x)$ de voorganger van x op het kortste pad van A naar x . [4]

step	N'	$D(B), p(B)$	$D(C), p(C)$	$D(D), p(D)$	$D(E), p(E)$
0	{A}	8,A	2,A	∞	∞
1					
2					
3					
4					

Een alternatief algoritme voor het bepalen van kortste (of goedkoopste) paden is het *distance-vector*-algoritme. In de rest van deze opgave richten we ons op dit algoritme, en in het bijzonder op node A, en het gezichtspunt dat node A heeft op het hierboven gegeven netwerk. In deelopgave 4d) t/m 4g) gaan we uit van het gebruik van het standaard *distance-vector*-algoritme, **zonder** *poisoned reverse*. Op tijdstip t_0 , na convergentie van het *distance-vector*-algoritme, heeft node A de volgende afstandtabel gecreëerd. Hierbij geeft de regel van A, A's eigen *distance vector*. De regels van B en C geven de *distance vectors*, zoals A ze heeft ontvangen van B, respectievelijk C.

A		Cost to				
		A	B	C	D	E
from	A (computed)	0	6	2	10	8
	B (received)	6	0	4	4	2
	C (received)	2	4	0	8	6

- d) Enige tijd later, op tijdstip t_1 ($t_0 < t_1$), ontdekt node A dat de kosten van de link van node A naar node B gewijzigd zijn van 8 naar 4. Als gevolg hiervan verandert de afstandtabel van node A. Geef alle waarden van de afstandtabel in de onderstaande tabel aan. [2]

A		Cost to				
		A	B	C	D	E
from	A (computed)					
	B (received)					
	C (received)					

- e) Weer enige tijd later, op tijdstip t_2 ($t_0 < t_1 < t_2$), ontvangt node A een nieuwe *distance vector* van node B. Deze luidt $D_B = [D_B(A), D_B(B), D_B(C), D_B(D), D_B(E)] = [4, 0, 8, 4, 2]$. Als gevolg hiervan verandert de afstandtabel van node A. Geef alle waarden van de afstandtabel in de onderstaande tabel aan. [2]

A		Cost to				
		A	B	C	D	E
from	A (computed)					
	B (received)					
	C (received)					

- f) Weer enige tijd later, op tijdstip t_3 ($t_0 < t_1 < t_2 < t_3$), ontvangt node A een nieuwe *distance vector* van node C. Deze luidt $D_C = [D_C(A), D_C(B), D_C(C), D_C(D), D_C(E)] = [2, 6, 0, 8, 10]$. Als gevolg hiervan verandert de afstandtabel van node A. Geef alle waarden van de afstandtabel in de onderstaande tabel aan. [2]

A		Cost to				
		A	B	C	D	E
From	A (computed)					
	B (received)					
	C (received)					

- g) Tenslotte, op tijdstip t_4 ($t_0 < t_1 < t_2 < t_3 < t_4$), ontdekt node A dat de kosten van de link van node A naar node B opnieuw gewijzigd zijn, en wel van 4 naar 100. Als gevolg hiervan verandert de afstandtabel van node A. Geef alle waarden van de afstandtabel in de onderstaande tabel aan. [2]

A		Cost to				
		A	B	C	D	E
From	A (computed)					
	B (received)					
	C (received)					

6) **Multiple access protocollen [13]**

Stel je een *broadcast* kanaal voor met een *data rate* $R = 24 \text{ Mbit/s}$ ($24 \cdot 10^6 \text{ bits/s}$), en $N = 8$ nodes. Je wordt gevraagd voor drie verschillende *multiple access* protocollen de *throughput per node* te berekenen. Daarbij is de slot of pakketgrootte steeds $S = 1500 \text{ bytes}$. Geef steeds de berekening van je antwoord en een korte motivatie.

In vraag a) t/m c) is er steeds slechts 1 node die continu data te versturen heeft.

a) Een TDM systeem met N tijdsloten per frame. [2]

b) Een *slotted Aloha* systeem, waarbij een node die data te versturen heeft altijd in een slot met kans $p = 1/4$ een pakket zal sturen. [2]

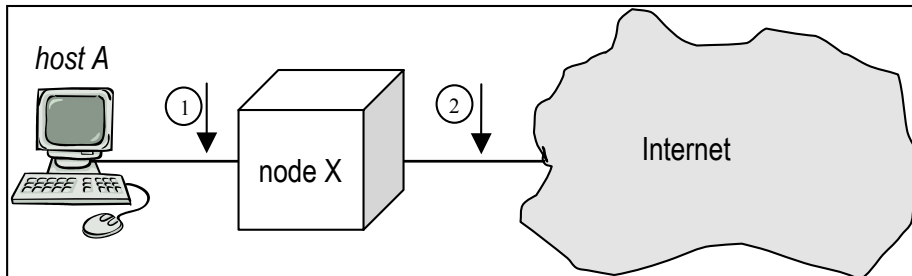
c) Een *polling* systeem, waarbij een node die gepolled wordt per beurt maximaal 1 pakket van grootte S mag versturen, en de tijd nodig voor het pollen van 1 node, de *polling delay* $t_{poll} = 0,25 \text{ ms}$ bedraagt. [2]

Beantwoord in vraag d) t/m f) dezelfde vraag (wat is de *throughput per node*) voor het geval $n \leq N$ nodes continu data te versturen hebben:

- d) Een TDM systeem met N tijdsloten per frame. [1]
- e) Een *slotted* Aloha systeem, waarbij een node die data te versturen heeft altijd in een slot met kans $p = 1/4$ een pakket zal sturen. [2]
- f) Een *polling* systeem, waarbij een node die gepolled wordt per beurt maximaal 1 pakket van grootte S mag versturen, en de tijd nodig voor het pollen van 1 node, de *polling delay* $t_{poll} = 0,25$ ms bedraagt. [2]
- g) Bij welke van de 3 hierboven genoemde *multiple access* protocollen treedt *statistical multiplexing* op? [2]

7) Adressering [10]

Beschouw de volgende configuratie. *Host A* is via een node met onbekende functionaliteit verbonden met het internet. Op de een of andere manier is het mogelijk de *header* informatie van de pakketten die host A **naar** het internet verstuurt te bekijken, en wel op 2 verschillende punten: punt 1 en punt 2. De belangrijkste *header* informatie is *Source IP Address* en *Destination IP Address*, *Source MAC Address* en *Destination MAC Address* en *Source Port Number* en *Destination Port Number*.



- a) Geef voor ieder van deze paren aan bij welke laag van de Internet protocol *stack* ze horen. [2]

Source MAC Address en *Destination MAC Address*:

Source IP Address en *Destination IP Address*:

Source Port Number en *Destination Port Number*:

- b) Stel dat host A het IP adres 129.7.254.131/22 heeft. Wat is de betekenis van de /22 in deze notatie? Wat is de network prefix van dit adres in binaire notatie? [2]

Stel dat we willen achterhalen wat node X is: een switch, een router, of een NAT. We nemen aan dat we van een pakket steeds maar één adresveld tegelijk kunnen bekijken, en dat het bij de observaties op punt 1 en 2 telkens met zekerheid gaat om hetzelfde pakket.

- c) Wanneer het *Destination MAC Address* op punt 1 en 2 **verschillend** is, **kan** node X dan een switch zijn? En een router? Een NAT? Omcirkel het juiste antwoord (ja/nee) en motiveer je antwoord. [1½]

Switch: ja / nee want:

Router: ja / nee want:

NAT: ja / nee want:

- d) Wanneer het Source IP Address op punt 1 en 2 **verschillend** is, **kan** node X dan een switch, router of NAT zijn? Omcirkel het juiste antwoord (ja/nee) en motiveer je antwoord. [1½]

Switch: ja / nee want:

Router: ja / nee want:

NAT: ja / nee want:

- e) Wanneer het Destination Port Number op punt 1 en 2 **gelijk** is, **kan** node X dan een switch, router of NAT zijn? Omcirkel het juiste antwoord (ja/nee) en motiveer je antwoord. [1½]

Switch: ja / nee want:

Router: ja / nee want:

NAT: ja / nee want:

- f) Wanneer het Source Port Number op punt 1 en 2 **gelijk** is, **kan** node X dan een switch, router of NAT zijn? Omcirkel het juiste antwoord (ja/nee) en motiveer je antwoord. [1½]

Switch: ja / nee want:

Router: ja / nee want:

NAT: ja / nee want: