

Tentamen Computerarchitectuur en Netwerken, 22 juni 2012

Omdat de vragen over computerarchitectuur en netwerken door verschillende docenten worden nagekeken wordt aangeraden om de antwoorden op de vragen over het onderdeel computerarchitectuur en het onderdeel netwerken op afzonderlijk antwoordpapier uit te werken. Wie dat nalaat riskeert een belangrijke vertraging bij de correctie van het ingeleverde werk.

Beantwoord de vragen over computerarchitectuur kort en bondig. Lange, uitgebreide antwoorden kunnen een reden zijn om een opgave als 'onvoldoende' te beoordelen.

Geef duidelijk aan bij welke vraag je antwoord hoort.

Succes!

Deel COMPUTERARCHITECTUUR

1. Wat zijn de niveaus die door Tanenbaum in de computerarchitectuur worden onderscheiden.

Geef bij elk niveau kort aan wat de functie van het niveau is.

2. Het geheugen van een computer gebruikt bytes (8 bits waarden), waaraan binnen in de geheugencellen nog eens 4 bits Hamming correctie-bits zijn toegevoegd. Er wordt gebruik gemaakt van de Hamming foutcorrectie- methode met even-parity.

Het foutcorrectiesysteem treft in een geheugencel de volgende waarde aan:

1 0 1 0 1 0 1 0 0 1 1 1

(bit 0 is de meest rechtse waarde, bit 11(decimaal!) de meest linkse).

Vragen:

- Wat is de waarde die op dit moment in de geheugencel is opgeslagen? Geef de decimale of hexadecimale waarde (de hexadecimale waarde svp beginnen met 0x).
(Toelichting: wanneer de waarde van deze geheugencel door b.v. een programma wordt opgevraagd, welke waarde zou dan, gegeven bovenstaand bit-patroon, worden ontvangen?)
- De geheugencel bevat een fout. Wanneer deze fout door het foutcorrectiesysteem is verholpen, welke waarde bevat de geheugencel dan? Geef de decimale of hexadecimale waarde (de hexadecimale waarde svp beginnen met 0x). Geef kort aan waarom deze correctie wordt uitgevoerd.

3. De volgende 20 bytes geven de inhoud van een binaire file die door een 32-bits little-endian computer zijn geschreven (de linker kolom zijn de byte-indices, de overige kolommen geven de inhoud van de bytes); letters zijn ascii-characters, getallen zijn numerieke byte-waarden). De bytes op adres 16 t/m 19 moeten worden geïnterpreteerd als een 32 bits waarde.

```

0000   h   e   l   l   o   32   w   o
0008   r   l   d   0   0   0   0   0
0016   4   3   2   1

```

Iemand schrijft een programma dat de tekst uit de eerste 16 bytes en vervolgens de binaire waarde die in de volgende 4 bytes is opgeslagen inleest.

Vragen:

- Wat is de numerieke waarde (decimale of hexadecimale waarde is OK, indien hexadecimaal laat de waarde dan beginnen met 0x) die in de bytes op adressen 16 t/m 19 is opgeslagen (NB: dus 1 waarde, niet 4 waarden)
- Wijzig de file zodanig dat een 64-bits big endian computer dezelfde tekst en dezelfde waarde zal lezen. Omdat de big endian computer een 64 bits architectuur heeft dient de waarde dus niet alleen omgezet te worden naar een big-endian architectuur, maar ook naar een 64-bits architectuur.

4. Beschouw een computer met 32-bits adressen. De adressen zijn opgesplitst in onderdelen, waarvan de 16 meest significante bits een 64 kB cache indiceren, gevolgd door 11 bits die de index van de te gebruiken 32 byte cache lijn aangeven, gevolgd door 3 bits die de de 4-byte 'woord' index binnen een 32 byte cache lijn aangeven, gevolgd door 2 bits die de byte index binnen het te gebruiken woord aangeven.

Geef de (decimale of desgewenst hexadecimale) waarden voor de cache, cache lijn, woord index en byte index voor adres 132015. Merk op: alle waarden zijn indices, waarbij 0 de laagste index waarde is.

Hint: 132015 is binair (spaties toegevoegd t.b.v. de leesbaarheid):

```

10 0000 0011 1010 1111
^          ^
meest          minst
significante   significante
bit           bit

```

5. Vrijwel alle programmeertalen kennen functies met argumenten. De argumenten worden doorgaans in omgekeerde volgorde op de stack geplaatst (dmv een 'push variabele-naam' instructie).

Beschouw de volgende functie-aanroep:

```
subtract(a, b)
```

Een instructie-set kent onder andere de instructie 'call functie-naam' om een functie aan te roepen. In deze instructie-set wordt EBP gebruikt om het stack-frame van de functie die op dat moment wordt uitgevoerd te bereiken, terwijl ESP altijd de index van het bovenste stack-element bevat.

Vragen:

- Hoe zal de aanroep van 'subtract(a, b)' worden omgezet naar assembler?
- Wat zijn de eerste twee assembler instructies die aan het begin van de functie 'subtract' zullen worden uitgevoerd? (Bedoeld worden de instructies die vrijwel altijd aan het begin van functies worden uitgevoerd.)

6. Een programma gebruikt een hash-tabel met 13 elementen. In de hash-tabel worden gegevens van personen opgeslagen. De namen van de eerste 10 personen die in de tabel worden opgeslagen, en de oorspronkelijke hash-waarden die voor deze namen zijn berekend, zijn:

Aart	4
Anke	2
Arnold	3
Astrid	6
Ingrid	8
Karel	12
Peter	5
Piet	6
Sarah	11
Wietze	3

Vragen:

- Op welke index posities (de kleinste index is 0, de grootste index is 12) bevinden zich de namen nadat bovengenoemde 10 namen in de gegeven volgorde in de tabel zijn geplaatst? Maak bij collisies gebruik van de add-the-hash-rehash (nieuwe hash-waarde = (laatst gebruikte hash-waarde + oorspronkelijke hash-waarde) modulo 13) methode om een nieuwe index waarde te vinden.
- In dit voorbeeld zijn er flink wat collisies. Wat is een eenvoudige manier om het aantal collisies te verminderen?

7. Ontwerp een omega switching network om 2 cpu's en 2 geheugens aan elkaar te koppelen.

Deel NETWERKEN

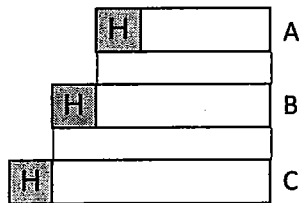
1. Het referentiemodel

a. Benoem de vijf lagen van de internet protocol-stack.

b. Geef voor onderstaande functies aan door welke laag (of lagen) ze worden uitgevoerd :

1. het bepalen van een pad door het netwerk
2. het versturen van berichten tussen processen op eindsystemen
3. foutdetectie en –correctie
4. aanpassing aan het kanaal zodanig dat een bit kan worden verzonden
5. het versturen van een pakket van een node naar een volgende node op het netwerkpad
6. multiplexing en de-multiplexing

c. Gegeven zijn de protocollen TCP, Ethernet, IP, HTML en HTTP. Plaats deze protocollen in een plaatje volgens onderstaand voorbeeld. (In dit voorbeeld gaat het om protocollen A, B en C).



d. Als vraag (c), maar nu voor de volgende protocollen : WiFi, DNS, IP en UDP.

2. fysieke laag

a. Een manier om de capaciteit van een communicatiekanaal te verhogen is het leggen van meerdere kabels of plaatsen van meerdere antennes. Maar dit is nogal kostbaar. Beschrijf op welke andere drie manieren je de capaciteit kunt verhogen.

b. Om informatie over een kanaal te transporteren kan modulatie efficiënte oplossing zijn; dit is afhankelijk van de fysieke eigenschappen zoals demping van het kanaal. Wat is precies modulatie ?

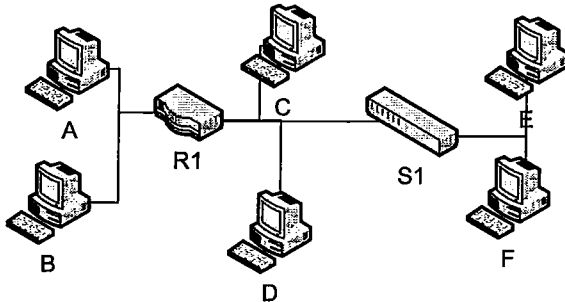
c. Beschrijf de drie basis-modulatiemethoden.

d. Een ADSL verbinding maakt gebruik van QAM-64 voor elk downstream kanaal. Een kanaal heeft een capaciteit van 5000 symbolen per seconde. Stel er zijn 100 downstream kanalen. Wat is de capaciteit van de downstream link in Mb/s ?

3. datalink laag en netwerklaag

a. Leg uit wat bij Random Access datalink protocollen (zoals Ethernet) wordt bedoeld met "collision domain".

Gegeven het volgende IP-netwerk bestaande uit hosts A t/m F, router R1 en Ethernet Switch S1.



Beschrijf

b. Hoeveel collision domains zijn er? Benoem de nodes van elk domein.

c. Hoeveel netwerken (subnetten) zijn er? Benoem de nodes in elk netwerk.

Stel host A wil een IP pakket sturen naar host F. Verder zijn IP A en IP F de IP adressen van A resp. F. Evenzo zijn MAC A en MAC F de MAC adressen van A resp. F.

d. Beschrijf wat R1 en S1 doen om dit te realiseren. Geef aan wat hierbij gebeurt met de adressen in het IP pakket en de adressen in het ethernet frame.

4. transport laag

UDP en TCP zijn invullingen van de transportlaag in het internet model. Bij de transportlaag gaat het om een logische verbinding tussen processen.

a. UDP en TCP gebruiken poortnummers om de bron en bestemming aan te duiden. Geef tenminste twee redenen waarom het gebruik van een proces ID (PID) niet voldoende is.

b. De header van TCP is veel groter dan die van UDP omdat TCP meer functies biedt. Welk belangrijke functie biedt TCP extra t.o.v. UDP? Noem twee velden die hierdoor wel voorkomen bij de TCP-header en niet bij UDP-header.

c. Het bekijken van films middels videostreaming is erg populair geworden. Zou je bij videostreaming kiezen voor UDP of TCP? Motiveer je antwoord.

