

Tentamen Computerarchitectuur en Netwerken, 3 juli 2009

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

Succes!

VRAGEN COMPUTER ARCHITECTUUR

=====

Beantwoord de vragen over computer architectuur kort en bondig. Lange, uitgebreide antwoorden kunnen een reden zijn om een opgave als niet voldoende te beoordelen.

Geef duidelijk aan bij welke vraag je antwoord hoort.

1. Wat zijn de niveaus die door Tanenbaum in de computer architectuur worden onderscheiden. Geef bij elk niveau kort aan wat de functie van het niveau is.
2. Waarom kan de informatie zoals die in een 'little endian' computer is opgeslagen niet eenvoudig worden omgezet naar een 'big endian' computer?

Stel een big endian computer heeft achtereenvolgens een 32 bits waarde, een tekst die wordt afgesloten door een 0-byte, en een 64 bits waarde als volgt opgeslagen in zijn geheugen (0x.. representeert een byte waarde, afzonderlijke characters representeren ascii-characters, elke waarde is opgeslagen in een byte):

(laag adres)

0x01, 0x02, 0x03, 0x04,
H, E, L, L, 0, 0x0,
0x05, 0x06, 0x07, 0x08, 0x09, 0x0a, 0x0b, 0x0c,

(hoog adres)

Converteer deze informatie naar een little endian computer. Geef het resultaat als een serie van 18 bytes van laag naar hoog adres.

3. De toegang tot de bus van de computer wordt vaak door een arbiter geregeld. Wat is (en wat is het verschil tussen) 'daisy chain' en 'priority chain' arbitrage? Hoe kan arbitrage zonder arbiter worden gerealiseerd? Geef bij dit laatste onderdeel een toelichting in de vorm van een tekening waarmee je het principe van arbitrage zonder arbiter duidelijk maakt.

4. Wanneer in een assembleertaal functies worden aangeroepen dan krijgt zo'n functie vaak argumenten mee en heeft zo'n functie vaak lokale variabelen. Veronderstel een computer waarvan het standaard data type een 32 bits 'int' is (de instructie 'push' plaatst een int op de stack, alle registers werken standaard met int waarden). We beschouwen een computer met de volgende eigenschappen:

EBP - het base pointer register
ESP - het stack pointer register

mov dst, src - assembler instructie om src toe te kennen aan dst
add dst, src - assembler instructie om src op te tellen bij dst
sub dst, src - assembler instructie om src af te trekken van dst
call functie - assembler instructie om 'functie' aan te roepen. Deze instructie plaatst tevens het return adres (RA) op de stack.
push src - assembler instructie om een waarde op de stack te plaatsen

De computer hanteert een standaard stack (groeïend van een hoog adres naar een laag adres) en kent 'immediate', 'register', en 'register indirect' als adresserings modi. In deze opgave is 'dst' altijd een register.

Wanneer een functie in de programmeertaal C wordt aangeroepen worden de argumenten van de functie in omgekeerde volgorde op de stack geplaatst (eerst het laatste argument, tenslotte het eerste argument).

Zet nu de volgende aanroep om in Assembler:

```
show(3, 5);
```

waarbij de relevante onderdelen van de functiedefinitie zijn:

```
void show(int a, int b)
{
    int v1;
    int v2;
    // ... (statements, hier niet van belang)
}
```

De code aan het begin van een functie heeft een karakteristieke vorm: een functie wordt gekenmerkt door een 'stack frame' dat als volgt is opgebouwd:

1. De oude base (frame) pointer wordt bewaard
2. De nieuwe base pointer wordt gedefinieerd
3. Op de stack wordt ruimte gemaakt voor de lokale variabelen.

Geef, bovenstaande beschrijving volgend, de assembler instructies die aan het begin van de functie 'show' worden uitgevoerd.

5. Noem vier methoden waarop op het niveau van de micro-architectuur de

prestaties van computers kan worden verbeterd. Geef voor elke methode kort aan wat de bedoeling/werking ervan is.

- 8 6. Construeer een omega switching netwerk voor 4 CPUs en 4 memories. Maak een tekening.

VRAGEN NETWERKEN

=====

- 8 7. Geef aan wat het belangrijkste argument is om sliding-window protocollen te gebruiken. Geef daarnaast aan wanneer go-back-n en wanneer selective repeat een passende methode van sliding window is om te gebruiken.
- 8 8. CSMA/CD wordt gebruik in Ethernet. Geef aan hoe CS en CD er voor zorgen dat de effective belasting op een Ethernet toeneemt
- 8 9. Routing werkt op de netwerklaag van het OSI-model. Leg uit wat de functie is van routing, en wat het fundamentele verschil is tussen distance-vector en linkstate-routing.
- 8 10. In de IPv4 -header is er ruimte gereserveerd voor de checksum. In de IPv6-header is deze checksum verdwenen: leg uit waarom dit is.