

Examenvragen 2023-2024

De bedoeling is om bij iedere vraag een kort, krachtig, volledig en juist antwoord te formuleren. De quoterings zal gebeuren op 2 (0 voor een fout antwoord, 1 voor een min of meer juist antwoord en 2 voor een volledig correct antwoord). Dus de moeilijkheid is eigenlijk om in zo weinig woorden/zinnen de vraag de beantwoorden. Let op, sommige details zijn weldegelijk belangrijk en dienen ook vermeld te worden. Zie daarom de lessen en eventueel uw eigen nota's.

Hoofdstuk 1

1. Wat zijn de voordelen van het gebruik van een compiler?
2. Wat zijn de nadelen van het gebruik van een compiler?
3. Wat zijn de voordelen van het gebruik van een interpreter?
4. Wat zijn de nadelen van het gebruik van een interpreter?

Hoofdstuk 2

5. Wat is de gedachte achter een CISC-architectuur?
6. Wat is het nadeel van een CISC-architectuur?
7. Wat is de gedachte achter een RISC-architectuur?
8. Waarom is de instructieset beperkt bij RISC?
9. Hoe wordt neerwaartse compatibiliteit bereikt tussen CISC en RISC?
10. Welke regels worden er aan RISC-instructies opgelegd?
11. Wat is een superscalaire architectuur? Hoe ziet de pipeline er hier uit en waarom zijn bepaalde aanpassingen doorgevoerd?
12. Op basis van wat werkt een GPU en hoe wordt dit bij een vectorprocessor gedaan?
13. Maak een duidelijk schets van de voorstelling van variabele **p** in little endian en big endian en leg uit wat het probleem juist is bij de omzetting van de ene naar de andere voorstelling.

```
typedef struct {  
    char str[4];  
    int leeftijd;  
} persoon;
```

....

Persoon p={"Wim",50};

14. Op basis van welk principe werken cachegeheugens en leg kort het principe uit? Geef hierbij enkele voorbeelden waaruit blijkt dat hetgeen je hiervoor schreef daadwerkelijk ook waar is.
15. Waarom werd RAID bedacht? Wat is m.a.w. het idee achter RAID?
16. Waarom is er bij RAID eigenlijk meer nood aan een redundante schijf?
17. Wat is de impact van de strip-grootte op de prestaties?
18. Bij RAID-4 is er een schrijfstraf...Die kan herleid worden tot twee leesopdrachten en twee schrijfoopdrachten. Gegeven een RAID-systeem met 4 disks; geef dan het bewijs dat dit kan met slechts 4 I/O-opdrachten ongeacht het aantal schijven.
19. Wat bedoelt men bij RAID-2 met mechanisch gesynchroniseerd en waarom is RAID-2 betrouwbaarder dan RAID-3?
20. Hoe komt het dat solid drives een write amplification factor hebben en gewoon klassieke magneetschijven niet?
21. Maak een schets van een floating gate MOSFET.
22. Wat is het probleem wanneer je DMA gebruikt in combinatie met cache-geheugens?
23. Geef stap voor stap aan hoe de CPU een I/O-opdracht geeft aan een I/O device en hoe het I/O device het antwoord terugstuurt.
24. Bij een bus waar ook de CPU zich op bevindt worden de prioriteiten onder de verschillende devices verdeeld. Welke prioriteit krijgt de CPU en waarom? Hoe noemt men dit fenomeen?
25. Wat is de functie van de embedded CPU bij een laserprinter?
26. Hoe worden grijstinten bekomen? Maak een schets en leg uit.
27. Waarom is het afdrukken van kleuren eigenlijk niet zo triviaal? Welke kleurmodellen worden er gebruikt?
28. Wat zijn de beperkingen van ASCII en hoe heeft men ASCII eigenlijk in eerste instantie uitgebreid?
29. Unicode werkt met 16-bit karakters. Welke problemen zijn hierbij voorgekomen en hoe heeft men die codering in eerste en enige instantie uitgebreid zonder de het aantal bits te verhogen?
30. Wat zijn de eigenschappen van UTF-8. Maak een schets van een sequentie van 3 bytes. Hoe worden ASCII-tekens voorgesteld?

Hoofdstuk 3

31. Maak een schets van een NAND-poort met CMOS-technologie. Hoeveel transistoren heb je nodig om een AND-poort te maken?
32. Maak een schets van een half-adder. Wat is het probleem? Maak vervolgens een schets van een full adder.

33. Wat is een ripple carry adder? De performantie kan je opdrijven door bv. een carry select adder te gebruiken. Hoe werkt dit?
34. Wat is het voordeel van het gebruik van de signalen RAS en CAS bij een geheugenchip? Wat is dan weer het voordeel van het gebruik van verschillende geheugenbanken?
35. Wat is meestal de breedte van de databus bij een CPU?
36. Waarom heb je een busdriver nodig voor het aansluiten van een I/O-controller op een bus? (twee redenen)
37. Maak een schets van een wired-or schakeling met twee devices die gebruikmaken van een open-collector busdriver. Waarom is dit wired-or?
38. Wat zijn de nadelen van een parallelle bus waardoor hedendaagse bussen seriële bussen zijn? Maak een schets om eventueel te verduidelijken.
39. Wat wordt er bedoeld met een full-handshake? Geef een voorbeeld.
40. Bespreek stap voor stap hoe een interrupt wordt afgehandeld bij gebruik van de 8259A prioriteitsinterruptcontroller.
41. Hoe werkt pipelining voor DDR3/4/5 SDRAM?
42. Bespreek memory mapped I/O en geef voor- en nadelen? Maak desnoods een schets.
43. Wat is isolated I/O, hoe werkt dit en welke zijn de voor- en nadelen? Maak desnoods een schets.
44. Wat is partiële adresdecodering en waarom is dit niet aangewezen?

Hoofdstuk 4

45. Wat is in zijn eenvoudigste vorm de definitie van een datapad?
46. Wat zijn de mogelijkheden om een geheugenaanvraag te doen via de MIC-1? Wanneer wordt een geheugenaanvraag gesteld en wanneer wordt het antwoord verwacht?
47. Wat is de datapadtiming van de MIC-1?
48. Waarom is timing op een datapad heel belangrijk?
49. Waarom moet de puls van de klok kort en krachtig zijn op het einde van een datapadcyclus?
50. Uit welke velden bestaat een micro-instructie en een micro-operatie? Wat is het verschil tussen beide?
51. Parallel aan de aansturing van het datapad wordt het adres van de volgende micro-instructie bepaald. Hoe gebeurt dit precies?
52. Geef het geheugenmodel van de IJVM en bespreek wat de verschillende delen bevatten en hoe ze gebruikt worden tijdens de uitvoer van een programma.
53. Hoe kan de uitvoeringstijd van ISA-instructies worden verbeterd?
54. Maak een schets van een IFU. Waarom is het belangrijk dat de C-bus constant in de gaten wordt gehouden?
55. Waarom zijn bij een gepipelined model van de MIC-2 drie latch registers toegevoegd?
56. Gegeven onderstaande MAL-code:

MAR = SP-1; rd

MAR = SP

H = MDR; wr

Maak van deze drie micro-instructies een gepipelineerde versie waarbij er 3 latch-registers (A, B en C) werden toegevoegd om de componenten onafhankelijk te doen werken. Welk probleem kom je hier tegen?

57. Er bestaan twee soorten lokaliteit, dewelke en geef bij beide duidelijke en juiste voorbeelden.
58. Veronderstel je hebt een direct mapped cache met 4096 rijen en cachelijnen van 128 bytes. Hoe kan je dan voor een 32-bit adres bepalen op welke rij in de cache je moet gaan zoeken? (de getallen kunnen worden aangepast). Wanneer zal je collisions krijgen (i.e. cachelijnen die naar dezelfde lijn verwijzen), geef de berekening van dat getal.
59. Veronderstel een 12-weg set associatieve cache met 8192 rijen en cachelijnen van 64 bytes, hoe kan je dan voor een 32-bit adres gaan bepalen op welk adres je moet gaan zoeken? Wat is het voordeel van het gebruik van een 12-weg set associatieve cache?
60. Hoe kan je schrijfoperaties naar cachelijnen gaan implementeren? Wat doe je voor louter schrijfoopdrachten?
61. Wat is het probleem bij de uitvoering van sprongopdrachten? Hoe kan dat worden opgelost zonder gebruik te maken van sprongvoorspelling? Wat is de meest eenvoudige vorm van sprongvoorspelling?
62. Veronderstel een geschiedenis tabel van 1024 lijnen en 64-bit instructielengtes. Hoe kan je dan bepalen op welke rij in de tabel je moet gaan zoeken? (de getallen kunnen worden aangepast). Wanneer zal je collisions krijgen (i.e. instructies die naar dezelfde lijn verwijzen), geef de berekening van dat getal.
63. Waarom zijn de prestaties met twee geschiedenisbits bij dynamische sprongvoorspelling beter. Geef een concreet voorbeeld om dit aan te tonen.
64. Gegeven een superscalaire architectuur die 2 instructies tegelijk kan hebben in de execute-fase en gegeven de volgende instructies die in volgorde worden uitgevoerd:

```
R0=R1+R2
R1=R3*R7
R1=R2*R3
R4=R1*R6
```

Welke problemen kom je tegen en waar? Hoeveel cycli zal dit vergen om in volgorde uit te voeren?

65. Gegeven een superscalaire architectuur die 2 instructies tegelijk hebben in de execute-fase en gegeven de volgende instructies die uit volgorde worden uitgevoerd:

```
R0=R1+R2
R1=R3*R7
R1=R2*R3
R4=R1*R6
```

Welke problemen kom je tegen en waar? Hoeveel cycli zal dit vergen om in volgorde uit te voeren?

66. Waarom is het zo dat de meeste systemen verplichten dat ISA-instructies in volgorde moeten stoppen?

67. Gegeven onderstaand codefragment (niet noodzakelijk dezelfde code):

```
void verdubbel_indien_gevonden(register int *tab, register int n, volatile int *getal){
    register int i=0;
    while (i<n && tab[i]!=getal)
        i++;
    if (i==n)
        *getal*=2;
}
```

Waarom is het onverstandig om `*getal*=2` naar voor te schuiven bij uitvoering? Welke problemen kan je tegenkomen en hoe kunnen ze opgelost worden?

68. Wanneer is het handig om poison bit bij een register te voegen? Geef een voorbeeld waar dit gebruikt wordt.

69. Bij de i7-pipeline wordt er gebruikgemaakt van store-to-load forwarding? Wat is dit en bespreek.

Hoofdstuk 5

70. Wat is geheugenalignering (geef een codevoorbeeld) en waarom is dit zo belangrijk?

Hoofdstuk 8

71. Wat is het idee achter een VLIW-CPU? Waarom presteert dit toch ondermaats?

72. Wat is predicated execution en wat is het voordeel?

73. Wat is fine-grained multithreading. Geef een voorbeeld met 4 threads en leg uit waarom dit niet altijd bruikbaar is?

74. Wat is coarse-grained multithreading en leg uit hoe dit werkt met 4 threads. Hoe kan dit eventueel nog iets worden geoptimaliseerd waardoor een dode cyclus kan worden overgeslagen?

75. Welke bronnen moeten bij multithreading sowieso worden gedupliceerd?

76. Wat is full-resource sharing, partitioned sharing en threshold sharing? Geef telkens voor- en nadelen.

Hoofdstuk 6

77. Aan welke vier randvoorwaarden moet ieder geheugenbeheersysteem voldoen?

78. Bespreek de werking van paginering (zonder virtueel geheugen). Wat is het verschil tussen paginering en vaste partitionering? Hoe gebeurt de adresvertaling bij paginering (maak een schets)?

79. Bespreek de werking van segmentatie (zonder virtueel geheugen)? Wat is het verschil tussen dynamische partitionering en segmentatie? Waarom wordt dit model voor de gebruiker bewust zichtbaar wordt gehouden. Geef een voorbeeld waar je handig gebruik kan maken van segmenten. Hoe gebeurt de adresvertaling bij segmentatie (maak een schets)?
80. Bespreek de stappen die moeten ondernomen worden wanneer bij adresvertaling wordt vastgesteld dat een deel van het proces zich niet in het geheugen bevindt?
81. Wat zijn de drie voordelen van het gebruik van virtueel geheugen? Wat is het nadeel van het gebruik van virtueel geheugen?
82. Welke twee parameters moet het besturingssysteem in de gaten houden om te zien of er bij het gebruik van virtueel geheugen te veel dan wel te weinig paginafouten optreden? Wat wordt er bedoeld met "thrashing"? Hoe kan het besturingssysteem oordeelkundig inschatten welke pagina's in de toekomst nodig zullen zijn en welke niet?
83. Wat is de ideale grootte van een pagina? Maak de afweging tussen de voor-en nadelen van kleine en grote pagina's en waarom men eigenlijk opteert voor paginagroottes van enkele kilobytes.
84. Wat is het verschil tussen een pagina-ingang bij een systeem met paginering zonder en met virtueel geheugen.
85. Wat is het probleem waarom men opteert voor systemen met dubbele paginering en een geïnverteerde paginatable?
86. Waarom heeft de MMU een TLB? Hoe werkt dit?
87. Wat is de werking van een systeem met een geïnverteerde paginatable?
88. Wat is de reden waarom systemen met virtueel geheugen niet meer uitsluitend gebruikmaken van segmentatie? Men lost dit op door gebruik te maken van systemen die gebruikmaken van paginering en segmentatie. Maak een schets hoe de adresvertaling hier verloopt.