

AFDELING A

Vraag 1 - Datavoorstelling

1.1 $[(x'y)(xz)]'$ ✓
 $= (x'y)' + xz$ ✓
 $= (x + y)' + xz$ ✓
 $= x(1+z) + y'$ ✓
 $= x + y'$ ✓

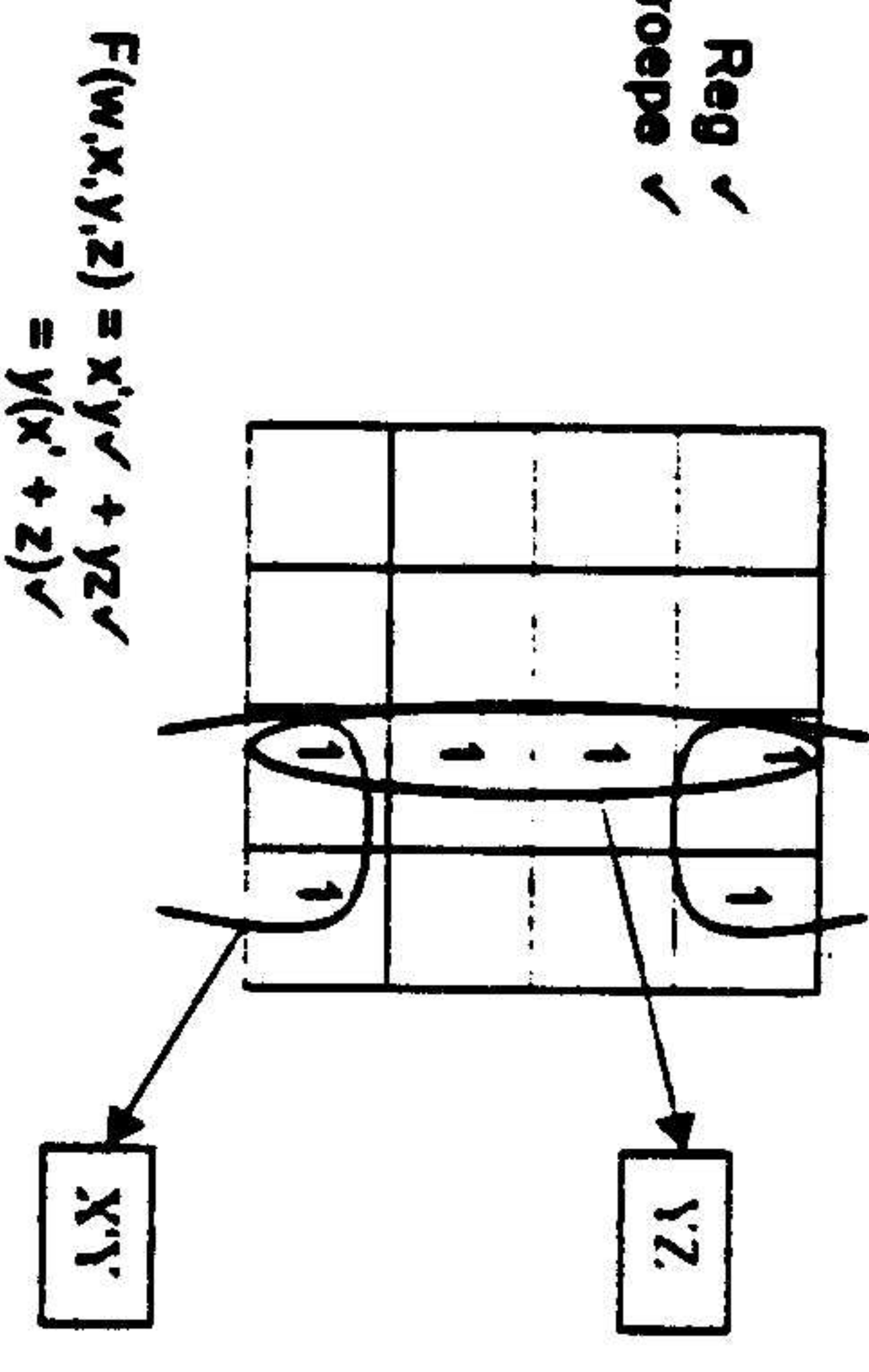
OF

$(x'y)(xz)'$ ✓
 $= x'y(x' + z)$ ✓
 $= x'y + x'yz'$ ✓
 $\rightarrow [x'y + x'yz']'$ ✓
 $= (x'y)'(x'yz)'$ ✓
 $= (x + y')(x + y' + z)$ ✓
 $= (x + y')(1 + z)$ ✓
 $= x + y'$ ✓

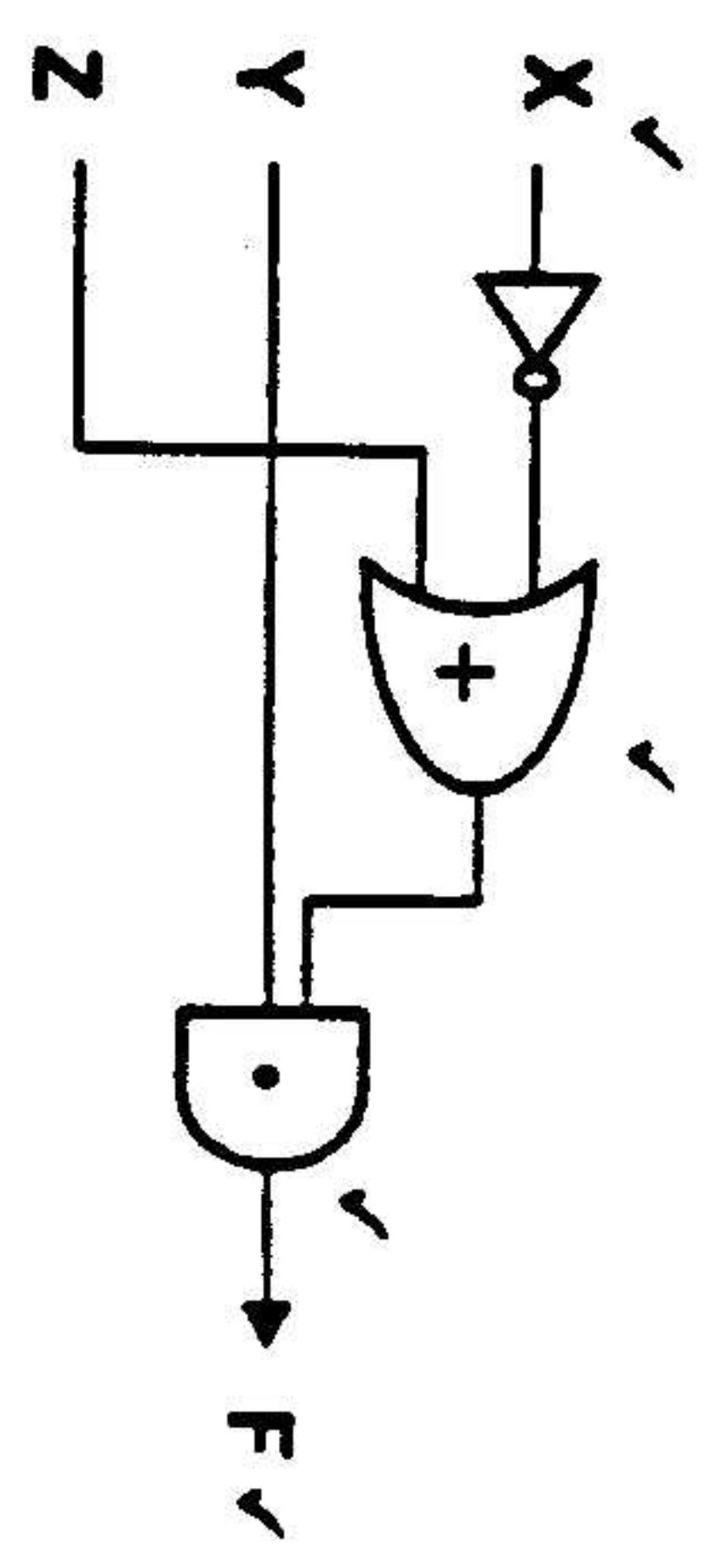
1.2 $x + y' = (x'y)'$ ✓

1.5.1 $F(w,x,y,z) = S(2,3,7,10,11,15)$ ✓

Reg ✓
 Groepe ✓



$F(w,x,y,z) = x'y' + yz'$ ✓
 $= y(x' + z)$ ✓



1.5.2

(6)

1.3.1 HO - Lewer som en oordrag van 2 bisse. ✓

(4)

1.3.2 VO - Lewer som en oordrag van 3 bisse. ✓

(2)

1.4 15 VO ✓ + 1 HO ✓

(2)

[19]

SECTION A

**Question 1
 Data representation**

1.1 $[(x'y)(xz)]'$ ✓
 $= (x'y)' + xz$ ✓
 $= (x + y)' + xz$ ✓
 $= x(1+z) + y'$ ✓
 $= x + y'$ ✓

OR

$(x'y)(xz)'$ ✓
 $= x'y(x' + z)$ ✓
 $= x'y + x'yz'$ ✓
 $\rightarrow [x'y + x'yz']'$ ✓
 $= (x'y)'(x'yz)'$ ✓
 $= (x + y')(x + y' + z)$ ✓
 $= x + y'$ ✓

1.2 $x + y' = (x'y)'$ ✓

(1)

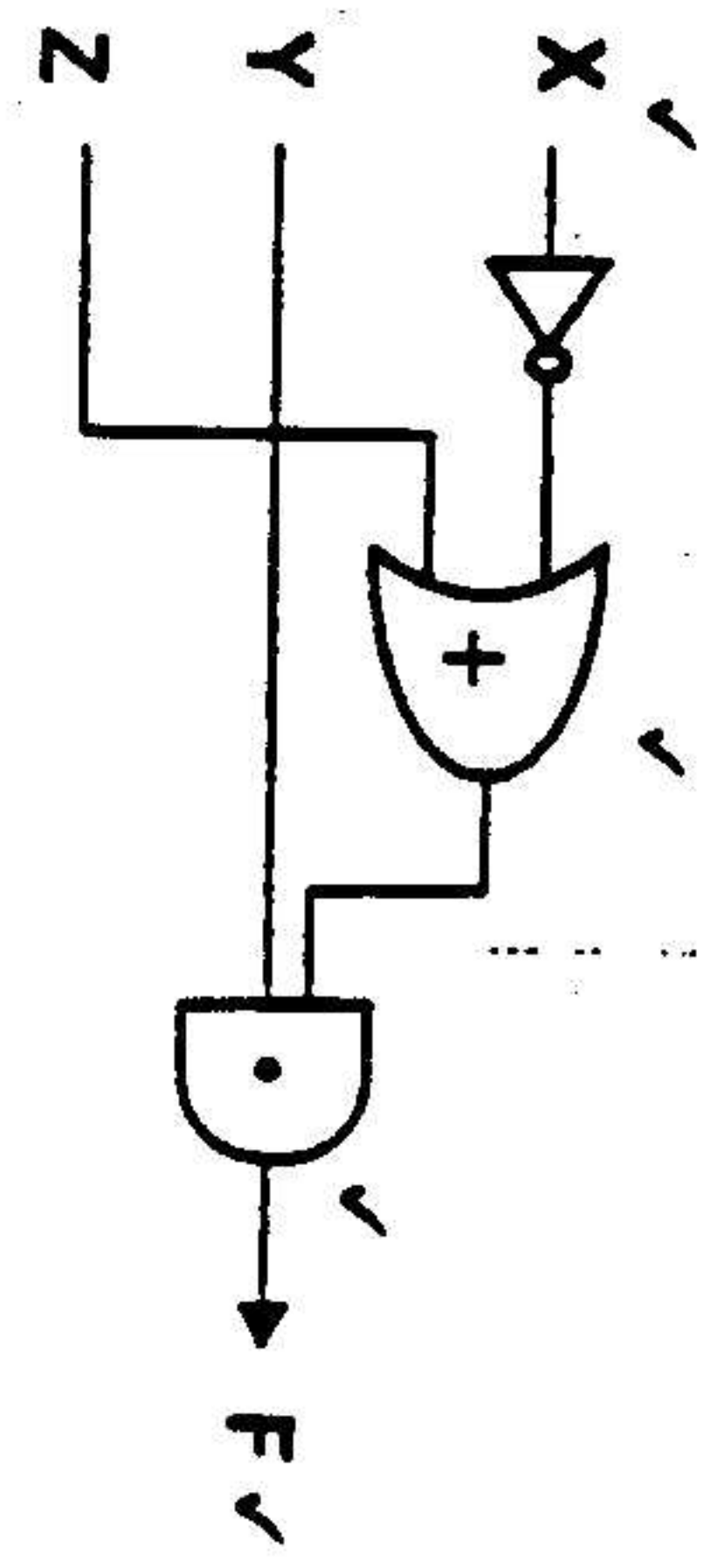
1.5.1 $F(w,x,y,z) = S(2,3,7,10,11,15)$ ✓

K-map ✓ ✓

$F(w,x,y,z) = x'y' + yz'$ ✓
 $= y(x' + z)$ ✓

1.5.2

(6)



1.3.1 HA - Gives sum and carry of 2 bits. ✓

(4)

1.3.2 FA - Gives sum and carry of 3 bits. ✓

(2)

1.4 15 FA ✓ + 1 HA ✓

(2)

[19]

X

GP 2002
Rekenaarsludie HG (Tweede Vraestel)
Vraestel X

Vraag 2
Argitektuur

- 2.1 Stelselklok**
2.1.1 reguleer alle aktiwiteite op die moederbord ✓ *Hyderberg* (1)
2.1.2 om tempo vir SVE aan te gee ✓ (1)
- 2.2 SVE**
2.2.1 RISC : eenvoudige masjieninstruksies ✓ wat vinnig uitgevoer kan word. ✓
CISC : Kombineer verskeie ✓ eenvoudige masjieninstruksies in een,
meer komplekse, instruksie. ✓ (4)
- 2.2.2 Verskeie instruksies by verskillende stadia van verwerking.
1 instruksie word per klokpuuls voltooi ✓ ✓ (2)
kasgeheue op verwerker self.
- 2.2.3 wanneer dit uit 2 pyplyne bestaan. ✓ (1)
- 2.3 Kasgeheue**
2.3.1 tempo van verwerker vinniger as primêre geheue ✓ daarom is
kasgeheue gebruik om die mees onlangse data te stoor vir vinnige
herwinning. ✓ (2)
- 2.4 Stelselbus**
2.4.1 data ✓ ; beheer ✓ ; adres ✓ (3)
2.4.2 Databus ✓ - meer instruksies en data ✓ kan op een slag ✓ vervoer word. (3)
- 2.5 Uitbreidingsgleuwe**
2.5.1 tempo te stadig ; wydte te nou ✓ (1)
2.5.2 PCI : VESA ; AGP ✓ ✓ (2)
2.5.3 kleur ; wydte ; kepe ✓ (1)
- 2.6 Poorte**
2.6.1 (a) COM - muis ; modem ; kommunikasiekabel (Laplink) ✓ (1)
(b) LPT1 - drukker ; scanner ; kommunikasiekabel (Laplink) ✓ (1)
(c) PS/2 - muis ; sleutelbord ✓ (1)
- 2.6.2 - seriepoort ✓ - tot 127 toestelle aan 1 poort ✓ (4)
- waarlik PnP ✓ - Hot swappable ✓
- 2.7 SIMM / DIMM-gleuwe**
2.7.1 om primêre geheue in te prop ✓ (1)
2.7.2 maklik om geheue uit te brei ✓ (1)
- [30]

X

GP 2002
Computer Studies HG (Second Paper)
... Paper X

Question 2
Architecture

- 2.1 System clock**
2.1.1 regulates all activities on the motherboard ✓ (1)
2.1.2 to give the tempo to be used by the CPU ✓ (1)
- 2.2 CPU**
2.2.1 RISC : Simple machine instructions ✓ that can be executed quickly. ✓
CISC : Combines various ✓ simple instructions into one, more complex
instruction ✓
1) CISC CPU executed complex instr. > RISC CPU.
- 2.2.2 Various instructions are at different stages of completion.
1 instruction per clock pulse completed ✓ ✓ (2)
cache memory on CPU itself.
- 2.2.3 when it consists of 2 pipelines. ✓ (1)
- 2.3 Cache memory**
2.3.1 tempo of the processor is faster than access to the primary memory ✓
hence cache memory was used to store the data that were most recently
used to be recalled quickly. ✓ (2)
- 2.4 System bus**
2.4.1 data ✓ ; control ✓ ; address ✓ (3)
2.4.2 Data bus ✓ - more instructions and data ✓ can be transferred at one
time ✓ (3)
- 2.5 Expansion slots**
2.5.1 tempo too slow ; width too small ✓ (1)
2.5.2 PCI : VESA ; AGP ✓ ✓ (2)
2.5.3 colour ; width ; gaps ✓ (1)
- 2.6 Ports**
2.6.1 (a) COM - mouse ; modem ; communication cable (Laplink) ✓ (1)
(b) LPT1 - printer ; scanner ; communication cable (Laplink) ✓ (1)
(c) PS/2 - mouse ; keyboard ✓ (1)
- 2.6.2 - serial port ✓ - up to 127 devices to 1 port ✓ (4)
- truly PnP ✓ - Hot swappable ✓ *(Laplink)*
- 2.7 SIMM / DIMM slots**
2.7.1 to plug in the primary memory ✓ (1)
2.7.2 for easy expansion of memory ✓ (1)
- [30]

X

Vraag 3
Stelselprogrammatuur

- 3.1.1 'n Versameling programme ✓ wat alle program- en apparatuuraktiwiteite monitor en beheer. ✓ (3)
 - 3.1.2 kort omkeertyd ✓ *bestuur geloue en foestelle* maksimum deurvoer ✓ hulpbronne effektief te benut ✓ (3)
 - 3.1.3 Windows 9X, Me, NT, 2000 : Linux : OS/2 ✓ ✓ ✓ (3)
 - 3.1.4 gebruik hardeskyfspasie ✓ as ekstensie ✓ van primêre geheue ✓ (3)
 - 3.1.5 buffers ✓ en spoolers ✓ (2)
 - 3.1.6 PDA ✓ (Personal Digital Assistants) (1)
 - 3.2 Aaneenskakeling : 'n Lêer word oor meer as 1 kluster ✓ gestoor en die verskillende klusters word logies saamtekaar geskakel ✓ (2)
Fragmentasie : Wanneer klusters van 1 lêer ✓ nie fisies na mekaar op 'n skyf lê nie ✓ is die lêer gefragmenteerd. (2)
 - 3.3.1 Geld onttrek (program beheer 'n aktiwiteit wat huidig plaasvind) ✓ (1)
 - 3.3.2 Bedienergesteunde netwerk ✓ (1)
 - 3.3.3 Meer as 1 aktiewe program oop in bv Windows 9X ✓ (1)
 - 3.4 I : om 'n bronprogram stelling-vir-stelling ✓ te vertaal en uit te voer ✓
K : om 'n bronprogram in sy geheel ✓ te vertaal ✓ na 'n objekprogram (masjientaal) ✓ (5)
 - 3.5 gratis ✓ : bronkode is beskikbaar vir publiek ✓ (2)
- [29]

X

Question 3
System Software

- 3.1.1 A set of programs ✓ to monitor and control ✓ all software and hardware activities. ✓ (3)
 - 3.1.2 fast turnaround time ✓ maximum throughput ✓ effective use of resources ✓ (3)
 - 3.1.3 Windows 9X, Me, NT, 2000 : Linux : OS/2 ✓ ✓ ✓ (3)
 - 3.1.4 uses space on hard disk ✓ as an extension ✓ of primary memory ✓ (3)
 - 3.1.5 buffers ✓ and spoolers ✓ (2)
 - 3.1.6 PDA ✓ (Personal Digital Assistants) *Designed for use on wireless communications devices, e.g. handheld computers, PDAs* (1)
Chaining : A file is stored over more than one cluster ✓ and the different *in-vehicle devices* clusters are logically chained together. ✓ (2)
Fragmentation : When the clusters of one file ✓ do not physically follow one another on the disk. ✓ (2)
 - 3.3.1 Withdrawal of money (program controls a current activity) ✓ (1)
 - 3.3.2 Server assisted network ✓ (1)
 - 3.3.3 More than 1 active program open in for example Windows 9X ✓ (1)
 - 3.4 I : to compile and execute ✓ the source code one statement at a time ✓
C : to compile and execute ✓ the source program as a whole ✓ to an object program (machine language) ✓ (5)
 - 3.5 available for free ✓ : source code is available to the public ✓ (2)
- [29]

Vraag 4

Datakommunikasie

- 4.1.1 Hubs ✓
 Netwerkkarte ✓
 UTP-kabels ✓
 Netwerkprogrammatuur ✓ (4)
- 4.1.2 Bedienergesteun : Novell ; Unix ; Windows NT / 2000 ✓
 Eweknie : Windows 9X / Me / 2000 ✓ (1)
- 4.1.3 Brug / Roeteerder ✓ (1)
- 4.1.4a Reëls vir datakommunikasie ✓ (1)
 4.1.4b TCP/IP of IPX ✓ (1)
- 4.1.5a Om te verseker dat elke stasie in 'n netwerk 'n gelyke kans het op
 versending ✓ (1)
 4.1.5b CSMA/CD ✓ (1)
 4.1.5c ROM op netwerkkarte ✓ (1)
- 4.1.6 Mikrorekenaar aan hoofraam ✓ (1)
- 4.2.1 Analooftlyn ✓ (1)
- 4.2.2 Intern ✓ en ekstern ✓ (2)
- 4.2.3 Ja ✓, 'n netwerk ✓ en internet connection sharing saggeware ✓ is nodig (3)
- 4.2.4 Satellietstelsels ✓ en WAP-fone ✓ (2)
- 4.3.1 Serie : een bit op 'n slag ✓ word versend
 Parallel : 'n Aantal bisse (bv. 8) op 'n slag ✓ word versend (2)
- 4.3.2 Baud : Tempo waarteen modem moduleer ✓
 bps : Die aantal bisse per sekonde ✓ (2)
- 4.3.3 HD : Tweerigtingkommunikasie ✓ maar in een rigting op 'n slag ✓
 VD : Tweerigtingkommunikasie ✓ gelyktydig in twee rigtings ✓ (4)
- 4.4.1 Lyn : Toegewyde pad lydens kommunikasie ✓
 Geskik vir luidende dataversending ✓ (2)
- 4.4.2 Boodskap : Geen toegewyde pad ✓
 Boodskap in geheel gestuur na bestemming ✓ (2)
- 4.5 Terwyl die data na die skyf geskryf word ✓ word dik oor verskillende skywe ✓
 gestoor saam met partitek ✓ sodat verlore data weer herwin kan word. ✓ (4)

[37]

Question 4

Data Communication

- 4.1.1 Hubs ✓
 Network cards ✓
 UTP cables ✓
 Network software ✓ (4)
- 4.1.2 Server assisted : Novell ; Unix ; Windows NT / 2000 ✓
 Peer-to-peer : Windows 9X / Me / 2000 ✓ (1)
- 4.1.3 Bridge / Router ✓ (1)
- 4.1.4a Laws for data communication ✓ (1)
 4.1.4b TCP/IP or IPX ✓ (1)
- 4.1.5a To ensure that every work station has an equal chance of sending a
 message. ✓ (1)
 4.1.5b CSMA/CD ✓ (1)
 4.1.5c ROM on network card ✓ (1)
- 4.1.6 Micro computer to main frame ✓ (1)
- 4.2.1 Analogue line ✓ (1)
- 4.2.2 Internal ✓ and external ✓ (2)
- 4.2.3 Yes ✓, A network ✓ and internet connection sharing software ✓ are needed ✓ (3)
- 4.2.4 Satellite dishes ✓ and WAP phones ✓ (2)
- 4.3.1 Serial : one bit at a time is being sent ✓
 Parallel : A number of bits (i.e. 8 bits) are being sent at a time ✓ (2)
- 4.3.2 Baud : Tempo at which modem modulates ✓
 bps : The number of bits per second ✓ (2)
- 4.3.3 HD : Two way communication ✓ but not simultaneously ✓
 FD : Two way communication ✓ simultaneously in both ways ✓ (4)
- 4.4.1 Circuit : Dedicated path during communication ✓
 Suitable for real time data ✓ (2)
- 4.4.2 Message : No dedicated path ✓
 Message sent as a whole to destination ✓ (2)
- 4.5 While data is being written to the disk ✓ it is stored over several disks ✓
 together with parity so that lost data can easily be reconstructed. ✓ (4)

[37]

Vraag 5
Sosiale Implikasies

- 5.1 Dit kombineer selfoortegnologie(draagbaar)✓ met PDA'draagbare rekenars✓ (2)
- 5.2.1 Ontwerp en stel netwerke op✓ en sorg dat die netwerk doeltreffend gebruik kan word✓ (2)
- 5.2.2 Sorg dat die apparatuur en programmatuur✓ van 'n webbediener✓ reg werk. (2)
- 5.3 Slegs as bewys kan word dat die bank finansiële skade✓ gelei het a.g.v. die "inbraak". (1)
- 5.4.1 Elektroniese aankope✓ oor die internet✓ (2)
- 5.4.2 Enkripsie-tegnologie✓ (1)
- 5.4.3 24 uur per dag✓ gemak van huis✓ nuutste pryse✓ geen kontakant hanteer✓ (4)
- 5.5 Kommunikasie (E-mail)✓ insaamel van inligting✓ (2)
- 5.6 Bv. Siggestreand : Groter skriftype of kleurkontras ✓✓ (2)

[18]
AFDELING A : [133]

Question 5
Socio-economic Implications

- 5.1 It combines cell phone technology(portable)✓ with PDA's/portable computers.✓ (2)
- 5.2.1 Designs and sets up networks✓ and make sure the network can be used effectively.✓ (2)
- 5.2.2 Ensures that the hardware and software✓ of a web server functions correctly.✓ (2)
- 5.3 Only if it can be proven that the banks financial loss✓ as a result of the break in. (1)
- 5.4.1 Electronic purchases via the internet.✓ (2)
- 5.4.2 Encryption technology✓ (1)
- 5.4.3 24 hours a day✓ comfort of own home✓ latest prices✓ no cash being handled✓ (4)
- 5.5 Communication (E-mail)✓ Gathering of information✓ (2)
- 5.6 E.g. Sight disabled : Use of larger fonts or colour contrasts ✓✓ (2)

[18]
SECTION A : [133]

X

AFDELING B

Vraag 6

- 6.1 Gemak van gebruik
 Buigsaamheid ✓✓✓
 Betroubaarheid ✓✓✓
 Doeltreffendheid ✓
 Gestidheid ✓ (4)
 - 6.2 Dit is die wyse waarop data in die rekenaar se geheue voorgestel word ✓ en wel op so 'n wyse dat toegang tot elke individuele element verkry kan word. ✓ (2)
 - 6.3.1 Program word opgedeel in kleiner dele ✓ wat elk 'n spesifieke probleem oplos ✓ (2)
 - 6.3.2 Programdele kan in eksterne eenhede gestoor word ✓ wat dan deur enige program gebruik kan word ✓ (2)
 - 6.3.3 Waardes word heen-en-weer ✓ tussen hoofprogram en subprogramme oorgegee ✓ (2)
 - 6.3.4 Veranderlikes wat slegs van toepassing is ✓ in die subprogram waarin dit verklaar is. ✓ (2)
 - 6.4.1 rewrite ✓ (4)
 - 6.4.2 gotoXY ✓ (4)
 - 6.4.3 ord ✓ (4)
 - 6.4.4 erase ✓ (4)
- [16]**

X

SECTION B

Question 6

- 6.1 Ease of use
 Compatibility ✓✓✓
 Reliability ✓
 Effectiveness ✓
 Suitability ✓ (4)
 - 6.2 The way data is represented internally in the computers memory ✓ in such a way that every element can be accessed individually. ✓ (2)
 - 6.3.1 Program is subdivided into smaller segments ✓ that each accomplish a certain task. ✓ (2)
 - 6.3.2 Program segments can be stored in external units ✓ that can be accessed by any program. ✓ (2)
 - 6.3.3 Values can be carried to an fro ✓ between main and subprograms. ✓ (2)
 - 6.3.4 Variables that are only applicable ✓ in the subprogram *in which* it was declared. ✓ (2)
 - 6.4.1 rewrite ✓ (4)
 - 6.4.2 gotoXY ✓ (4)
 - 6.4.3 ord ✓ (4)
 - 6.4.4 erase ✓ (4)
- [16]**

X

Vraag 7

```

repeat
  writeln('Gik die lidnummer in');
  readln(Lidnom);
  geldig := (uppercase(Lidnom[1]) in ['A'..'Z']) and (length(Lidnom) = 6);
  if not geldig then
    writeln('Ongeldig, probeer weer');
  until geldig;

Lus
  Toevoer
  Hoof + kleinletters
  Toets 1o karakter
  Toets lengte
  If-stelling binne lus met voorwaarde
  Foutboodskap
  Staakvoorwaarde
  Versamelingnotasie

```

[10]

Vraag 8

8.1

```

(1) reset(Teksleer);
(2) readln(Teksleer,Lyn);
(3) plek := pos(' ',Lyn);  plek = 1
(4) WNom := copy(Lyn,1,plek - 1);
(5) JSalStr := copy(Lyn,plek + 1,length(Lyn));
(6) val(JSalStr,JSalaris,Kode);
(7) MSalaris := JSalaris/12;
(8) writeln(WNom, '8,MSalaris:8:2');

```

(15)

8.2

```

var
  Teksleer      : text;
  Lyn,WNom,JSalStr : string;
  JSalaris,MSalaris : real;
  Kode          : integer;

```

[19]

X

Question 7

```

repeat
  writeln('Please enter the member number');
  readln(MemNum);
  Valid := (uppercase(MemNum[1]) in ['A'..'Z']) and (length(MemNum) = 6);
  if not Valid then
    writeln('Invalid, please try again. ');
  until Valid;

Loop
  Input
  Upper + lower case
  Test first character
  Test length
  If statement inside loop with condition
  Error message
  Condition for stop
  Set notation

```

[10]

Question 8

8.1

```

(1) reset(Textfile);
(2) readln(Textfile,Line);
(3) Place := pos(' ',Line);
(4) ENum := copy(Line,1,Place - 1);
(5) ASalStr := copy(Line,Place + 1,length(Line));
(6) val(ASalStr,ASalary,Code);
(7) MSalary := ASalary/12;
(8) writeln(ENum, '8,MSalary:8:2');

```

(15)

8.2

```

var
  Textfile      : text;
  Line,ENum,ASalStr : string;
  ASalary,MSalary : real;
  Code          : integer;

```

[19]

X

Vraag 9

```

untl' Vraag_9:
interface'
type
  SkikType = array[1..50] of Integer; ✓(Plaek)
procedure Someer(GetalSkik : SkikType ; var Som : Integer); ✓(met
  parameters)
Implementation'
procedure Someer; ✓(sondar parameters)
var
  I : Integer;
begin
  Som := 0; ✓
  for I := 1 to 50 do
    Inc(Som, GetalSkik[I]);
  end;
end. ✓

```

[9]

X

Question 9

```

untl' Question_9:
interface'
type
  ArrType = array[1..50] of Integer; ✓(Place)
procedure Add(NumArr : ArrType ; var Sum : Integer); ✓(with
  parameters)
Implementation'
procedure Add; ✓(without parameters)
var
  I : Integer;
begin
  Sum := 0;
  for I := 1 to 50 do
    Inc(Sum, NumArr[I]);
  end;
end. ✓

```

[8]

Vraag 10

Procedure BerekenBedrag(Sitplek : Texter; var Bedrag : real);
var
I : char;
J : Integer;

```

begin
  Bedrag := 0;
  for I := 'A' to 'D' do
    for J := 1 to 15 do
      if (sitplek[I,J] = true) then
        Bedrag := Bedrag + 20;
      for I := 'E' to 'J' do
        if (sitplek[I,J]) then
          Bedrag := Bedrag + 50;
        end;
      end;
    end;
  end;
end;

```

```

begin
  Bedrag := 0;
  for I := 'A' to 'D' do
    for J := 1 to 15 do
      case I of
        'A'..'D' : if sitplek[I,J] then
          Bedrag := Bedrag + 20;
        'E'..'J' : if sitplek[I,J] then
          Bedrag := Bedrag + 50;
        end;
      end;
    end;
  end;
end;

```

```

begin
  Bedrag := 0;
  for I := 'A' to 'J' do
    for J := 1 to 15 do
      if sitplek[I,J] then
        if I in ('A'..'D') then
          Bedrag := Bedrag + 20;
        else
          Bedrag := Bedrag + 50;
        end;
      end;
    end;
  end;
end;

```

[12]
AFDELING B : [67]
TOTAAL : 200

Question 10

Procedure CalculatesSales(Seat : Texter; var Amount : real);
var
I : char;
J : Integer;

```

begin
  Amount := 0;
  for I := 'A' to 'D' do
    for J := 1 to 15 do
      if (seat[I,J] = true) then
        Amount := Amount + 20;
      for I := 'E' to 'J' do
        if (seat[I,J]) then
          Amount := Amount + 50;
        end;
      end;
    end;
  end;
end;

```

```

begin
  Amount := 0;
  for I := 'A' to 'D' do
    for J := 1 to 15 do
      case I of
        'A'..'D' : if seat[I,J] then
          Amount := Amount + 20;
        'E'..'J' : if seat[I,J] then
          Amount := Amount + 50;
        end;
      end;
    end;
  end;
end;

```

```

begin
  Amount := 0;
  for I := 'A' to 'J' do
    for J := 1 to 15 do
      if seat[I,J] then
        if I in ('A'..'D') then
          Amount := Amount + 20;
        else
          Amount := Amount + 50;
        end;
      end;
    end;
  end;
end;

```

[12]
SECTION B : [67]
TOTAL : 200