

1 ONTWERPTECHNIEKEN

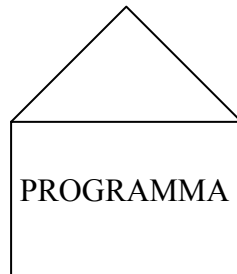
Bouwstenen

Methode voor het ontwerpen van een programmastructuur

{
Sequentie
Selectie
Iteratie

1.1 *Achtergrond*

1.1.1 Ruimtelijk denken in plaats van sequentieel denken



architect
ONTWERP

aannemer
UIVOERING



RUIMTELIJK DENKEN

SEQUENTIEEL



Programmeur

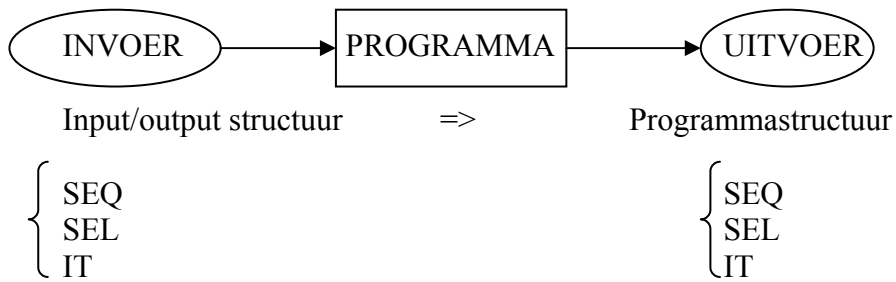
1.1.2 **Probleemstructuur = Programmastructuur**

PROBLEEM => PROGRAMMA
Dus Probleemstructuur => programmastructuur

Dit betekent dat men in het programma de verschillende elementen van het probleem moet zien.

GEVOLG: Probleemwijziging => wijziging in programmastructuur




PROBLEEM: Waar haalt men probleemstructuur?
DATA SCHOOL



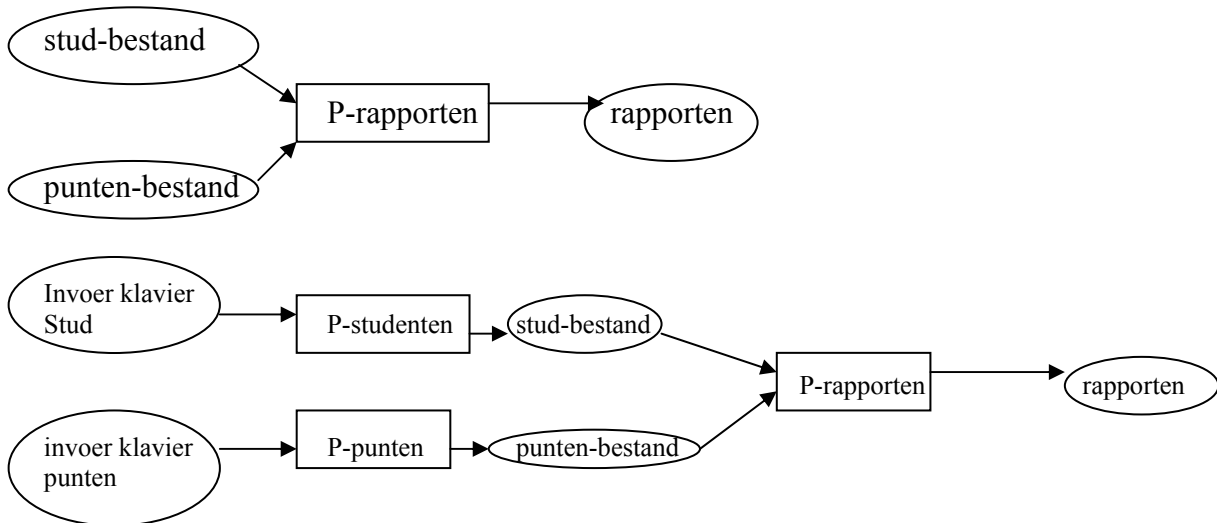
1.2 Afbeeldingsvormen

1.2.1 Systeemnetwerkdigram (SND)

SND voorstelling van het systeem = verzamelen van programma's, gegevensstromen en hun koppelingen

programma → rechthoek 
 gegevensstroom → cirkel 
 koppeling → pijl 

Voorbeeld

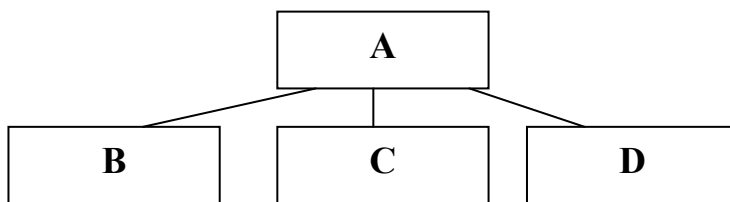


1.2.2 Structuurdiagram - structuurtekst

Zowel de structuur van de datastromen als deze van het programma worden op eenzelfde grafische wijze voorgesteld, nl. door middel van blokken die in een boomvormige structuur georganiseerd zijn. Ieder elementair structuurtype wordt voorgesteld door 2 niveaus: Het bovenste niveau heeft altijd 1 enkel blok, het onderste 1 of meer blokken.

1.2.2.1 Structuurtype SEQUENTIE

a) Structuurdiagram (grafische voorstelling)



Lees: A is een sequentie, bestaande uit element B, gevolgd door element C en gevolgd door element D.

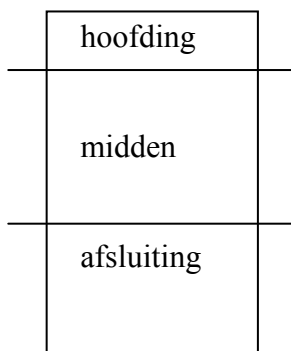
Opm: De volgorde van de elementen is van belang

b) Structuurtekst (weergave onder vorm van tekst) (pseudo-code)

```
A seq  
  B;  
  C;  
  D;  
A end
```

Voorbeeld: bij volgende 2 vbn stemt de programmastructuur overeen met de datastructuur.

1) *Datastructuur*



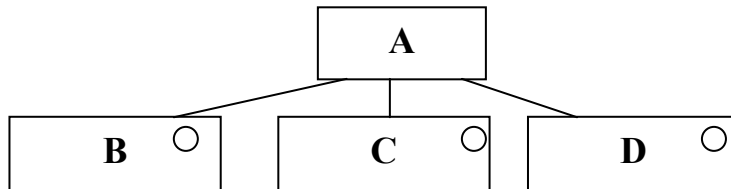
Een factuur bestaat uit een hoofding, gevolgd door een midden, gevolgd door een afsluiting.

2) *Programmastructuur*

Een programma om 1 enkele factuur te maken, bestaat uit een programma-onderdeel om de hoofding te maken, gevolgd door een programma-onderdeel om het midden te maken, gevolgd door het programma-onderdeel om de afsluiting te maken.

1.2.2.2 Structuurtype SELECTIE

a) Structuurdiagram (grafische voorstelling)



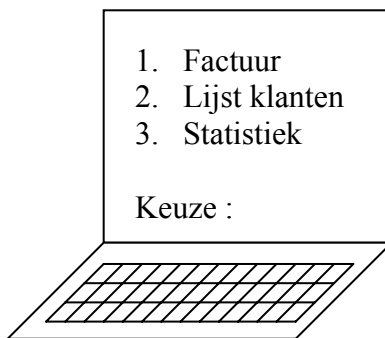
Lees: A is een selectie van ofwel element B, ofwel element C, ofwel element D
Opm : De volgorde van de elementen op het onderste niveau is van geen belang

b) Structuurtekst (weergave onder vorm van tekst) (pseudo-code)

```
A sel  
  B;  
A alt;  
  C;  
A alt;  
  D;  
A end;
```

Voorbeeld

1) Datastructuur



Een inputrekwest voor een programma is ofwel een rekwest om een factuur te maken, ofwel een rekwest om de lijst van de klanten te tonen ofwel een rekwest om een verkoopsstatistiek af te drukken.

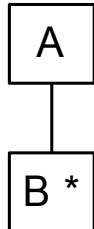
2) Programmastructuur

Het programma moet ofwel een factuur maken ofwel een klantenlijst tonen ofwel een verkoopsstatistiek afdrukken.

Opm: een speciale soort selectie is deze waarbij één van de onderste blokken leeg is; een leeg blok wordt aangeduid door er een streepje in te zetten.

1.2.2.3 Structuurtype ITERATIE

a) Structuurdiagram (grafische voorstelling)



Lees: A is een iteratie bestaande uit de herhalende elementen B, waarvan er dus bij definitie 0, 1 of meerdere kunnen zijn, naargelang de omstandigheden

b) Structuurtekst (weergave onder vorm van tekst) (pseudo-code)

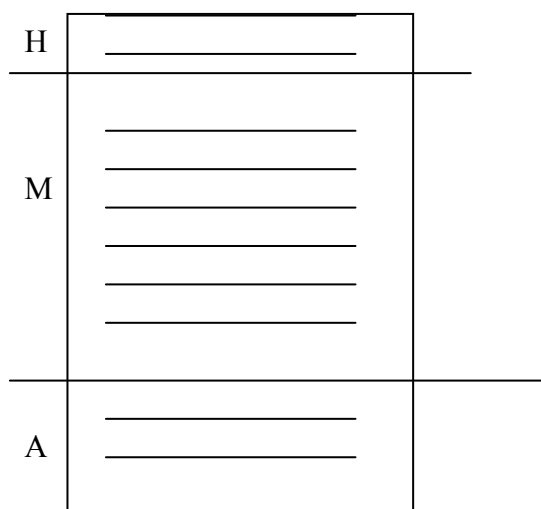
A itr

B;

A end

Voorbeeld

1) *Datastructuur*: een factuurmidden bestaat uit factuurlijnen.



2) *Programmastructuur*:

In een facturatieprogramma komt een routine voor om een factuurmidden te maken. Deze routine roept een andere routine op voor elke factuurlijn die moet gemaakt worden. Doorgaans bestaat een datastructuur en dus ook een programmastructuur uit meer dan 2 niveaus. Op ieder van deze niveaus komen dan 1 of meerdere iteraties, selecties en sequenties voor. Aldus verkrijgen we voor beide structuren een boomvorm, waarbij de blokken van het onderste niveau de bladeren zijn.

Voorbeeld:

1.3 ONTWERPPROCEDURE

De JSP-ontwerpprocedure bestaat uit 6 stappen

- Netwerkstap (systeemnetwerkdigram)
- Gegevensstap (datastructuren)
- Programmastap (programmastructuur)
- Opdrachten en condities-stap (Opdrachten en condities)
- Tekststap (pseudocode)
- Implementatiestap (Implementatie – code)

We zullen deze stappen voorstellen en illustreren met een zeer eenvoudig voorbeeld.

Opgave 10

Ontwerp een COBOL-programma dat de verkoopprijs van meerdere artikels berekent

Artikelgegevens	artikelnummer	N	6 pos (9(6))
	aankoopprijs	N	4 pos (geheel) (9(4))
	winstpercentage	N	99V99 of 99
	BTW-percentage	N	99V99 of 99
	Verkoopprijs	N	9(4)V99 of 9999

$\text{verkoopprijs} = \text{aankoopprijs} * (1 + \text{winstpercentage} / 100) * (1 + \text{BTW-Percentage} / 100)$

SCHERM-1

HOGESCHOOL GENT		Datum
BEREKENEN VERKOOPPRIJS		
Artikelnummer : Aankoopprijs : Winstpercentage : BTW-percentage :		
Verkoopprijs :		
Naam	Opgave 10	

Het programma moet stoppen als men als artikelnummer 0 ingeeft.

Dialogprogramma (interactief programma)



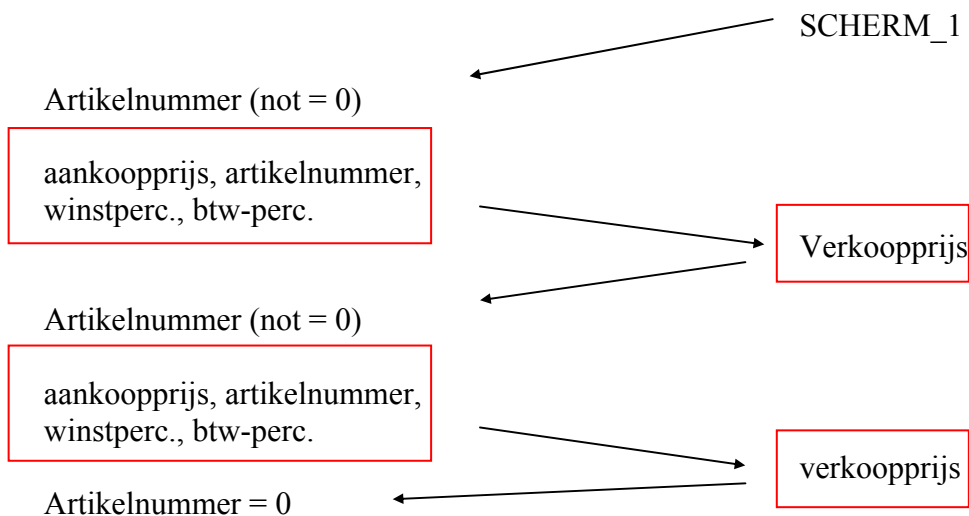
BATCH-programma

Dialog tussen gebruiker en programma

INPUT (gebruiker)

OUTPUT (programma)

Artikelgegevens



1) Systeemnetwerkdigram

Teken een SND waarbij het te ontwerpen programma voorgesteld wordt door een rechthoek en waarbij de input- en outputstromen weergegeven zijn door cirkels.

Systeemnetwerkdigram



2) Datastructuren - Gegevensstap

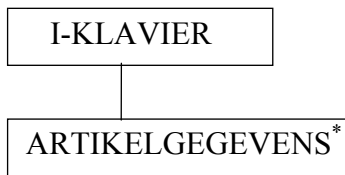
Teken een structuurdiagram voor iedere datastroom.

Merk op dat de inputstroom (=I-klavier) bestaat uit een herhaling van artikelgegevens.

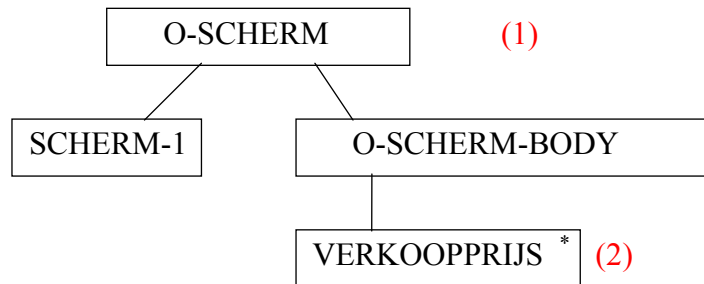
Merk op dat de outputstroom (=O-scherm) bestaat uit scherm-1 gevolgd door een herhaling van verkoopprijzen (=O-scherm-body)

Datastructuren

invoer:



uitvoer:



3) Programmastructuur

Hier stelt men de programmastructuur op als volgt. Meng de verschillende datastructuren tot 1 enkele programmastructuur. Hierbij moet men er zorg voor dragen dat ieder element in de datastructuren overeenkomt met 1 element in de programmastructuur en dat 2 overeenkomstige elementen van de datastructuren slechts vertegenwoordigd worden door 1 en hetzelfde element in de programmastructuur. We moeten dus eerst de overeenkomst(en) onderzoeken van elementen in de input-structuur met elementen in de output-structuur. Twee elementtypes uit beide structuren (1 uit input en 1 uit output) komen overeen als er van dat elementtype in beide structuren evenveel voorkomen, en als daarenboven de elementen die tot dat type behoren in beide structuren in dezelfde volgorde voorkomen.

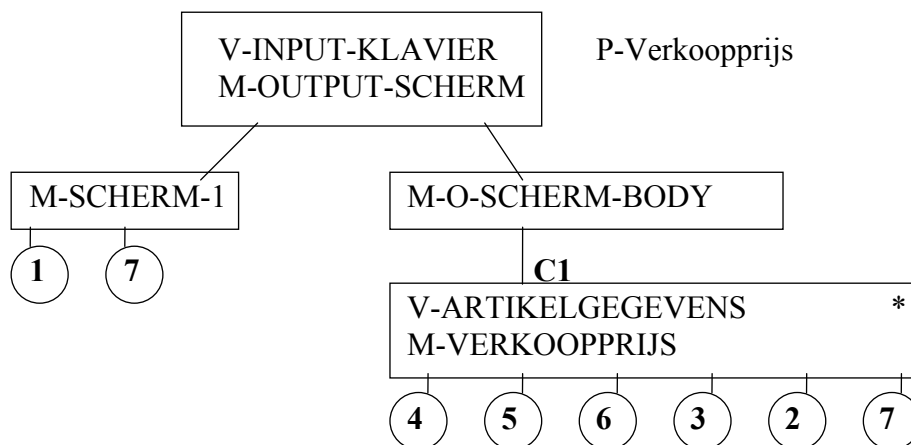
Programmastructuur

(1) INPUT-KLAVIER \Leftrightarrow OUTPUT-SCHERM

- beide komen 1 maal voor, dus ook zelfde volgorde

(2) ARTIKELGEGEVENS \Leftrightarrow VERKOOPPRIJS

- er zijn evenveel verkoopprijzen als artikelgegevens, zelfde volgorde



V = VERWERK (INPUT)

M = MAAK (OUTPUT)

4) Opdrachten en condities-stap

Vooreerst stellen we ons de vraag welke opdrachten dit programma zal moeten uitvoeren en welke condities moeten worden gebruikt (selecties en iteraties). Van beide maken we een lijst. Aan iedere opdracht en aan iedere conditie geven we een nummer. De 2^{de} stap van deze fase bestaat erin de opdrachten en de condities een plaats te geven binnen de programmastructuur.

! Merk op dat de operatie '(7) voerin (klavier,artikelnummer)' tweemaal voorkomt. Een eerste maal bovenaan en een tweede maal onderaan. Dit noemt men het *principe van het vooruitlezen*. Het principe van het vooruitlezen is noodzakelijk omdat we met iteraties werken, waar bij het begin van de iteratie de conditie C1 getest wordt. Testen van deze conditie betekent dat we nagaan of de waarde 0 werd ingelezen. Om deze test te kunnen doen moet er vooraf ten minste 1 artikelnummer zijn gelezen.

Opdrachten en condities

C1) ZOLANG (artikelnummer ≠ 0) DOE...

- 1) voeruit (scherm, SCHERM-1)
- 2) voeruit (scherm, verkoopprijs)
- 3) verkoopprijs ← aankoopprijs * (1+ winstpercentage / 100) * (1+ btwpercentage / 100)
- 4) voerin (klavier, aankoopprijs)
- 5) voerin (klavier, winstpercentage)
- 6) voerin (klavier, btwpercentage)
- 7) voerin (klavier, artikelnummer)

5) Tekstfase

Het programmastructuurdiagram wordt nu getransformeerd naar een structuurtekst (pseudo-code)

Tekststap

P-oef10

BEGIN

 M-SCHERM-1

 M-O-SCHERM-BODY

EINDE

M-SCHERM-1

BEGIN

 Voeruit (scherm, SCHERM-1)

 Voerin (klavier, artikelnummer)

EINDE

M-O-SCHERM-BODY

BEGIN

 ZOLANG (artikelnummer ≠ 0)

DOE V-artikelgegevens-M-verkoopprijs
EINDE-ZOLANG-DOE
EINDE

V-artikelgegevens-M-verkoopprijs
BEGIN
 voerin (klavier, aankoopprijs)
 voerin (klavier, winstpercentage)
 voerin (klavier, btwpercentage)
 verkoopprijs ← aankoopprijs (1+winstpercentage/100) (1+btwpercentage/100)
 voeruit (scherm, verkoopprijs)
 voerin (klavier, artikelnummer)
EINDE

6) Implementatiestap

Coderen van het programma in de programmeertaal, hier COBOL

Opgave 11 : Menugestuurd programma -

Ontwerp een Cobol-programma dat van artikels, de verkoopprijs, de aankoopprijs of het winstpercentage berekent al naargelang de ingevoerde keuze. Het programma wordt gestuurd door het volgend menu:

SCHERM-0

HOGESCHOOL GENT		Datum
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">BEREKENENINGEN</div>		
<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: 0 auto;"><p>1. Verkoopprijs :</p><p>2. Aankoopprijs :</p><p>3. Winstpercentage :</p> <p>0. EINDE :</p></div>		
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">Keuze :</div>		
Naam		Opgave 11

Vervolgens komt men in 1 van de volgende schermen:

SCHERM-1

HOGESCHOOL GENT		Datum								
BEREKENENEN VAN VERKOOPPRIJS										
<table border="1"><tr><td>Artikelnummer</td><td>:</td></tr><tr><td>Omschrijving</td><td>:</td></tr><tr><td>Aankoopprijs</td><td>:</td></tr><tr><td>Winstpercentage</td><td>:</td></tr></table>			Artikelnummer	:	Omschrijving	:	Aankoopprijs	:	Winstpercentage	:
Artikelnummer	:									
Omschrijving	:									
Aankoopprijs	:									
Winstpercentage	:									
<table border="1"><tr><td>Verkoopprijs</td><td>:</td></tr></table>			Verkoopprijs	:						
Verkoopprijs	:									
<table border="1"><tr><td>Druk op een toets...</td></tr></table>			Druk op een toets...							
Druk op een toets...										
Naam		Opgave 11								

$$\text{verkoopprijs} = \text{aankoopprijs} * (1 + \text{winstpercentage} / 100)$$

SCHERM-2

HOGESCHOOL GENT		Datum								
BEREKENENEN VAN AANKOOPPRIJS										
<table border="1"><tr><td>Artikelnummer</td><td>:</td></tr><tr><td>Omschrijving</td><td>:</td></tr><tr><td>Verkoopprijs</td><td>:</td></tr><tr><td>Winstpercentage</td><td>:</td></tr></table>			Artikelnummer	:	Omschrijving	:	Verkoopprijs	:	Winstpercentage	:
Artikelnummer	:									
Omschrijving	:									
Verkoopprijs	:									
Winstpercentage	:									
<table border="1"><tr><td>Aankoopprijs</td><td>:</td></tr></table>			Aankoopprijs	:						
Aankoopprijs	:									
<table border="1"><tr><td>Druk op een toets...</td></tr></table>			Druk op een toets...							
Druk op een toets...										
Naam		Opgave 11								

$$\text{aankoopprijs} = \text{verkoopprijs} / (1 + \text{winstpercentage} / 100)$$

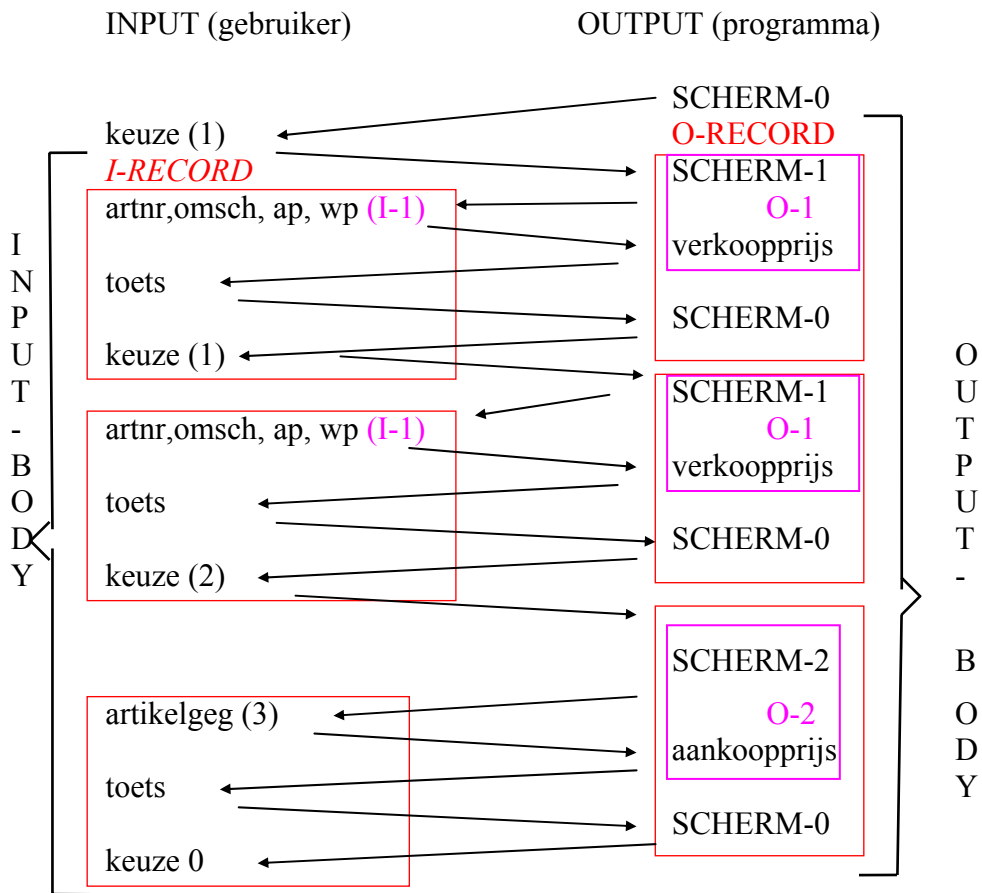
SCHERM-3

HOGESCHOOL GENT		Datum
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">BEREKENENEN VAN WINSTPERCENTAGE</div>		
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> Artikelnummer : Omschrijving : Aankoopprijs : Verkoopprijs : </div>		
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">Winstpercentage :</div>		
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;">Druk op een toets...</div>		
Naam		Opgave 11

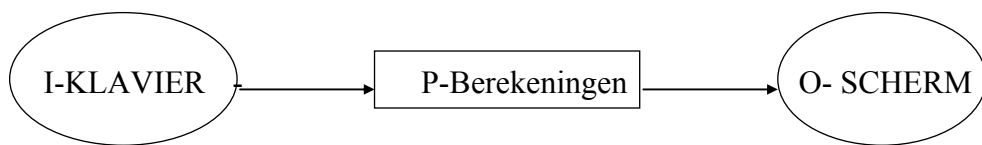
$$\text{winstpercentage} = \frac{(\text{verkoopprijs} - \text{aankoopprijs}) * 100}{\text{aankoopprijs}}$$

artikelgegevens:	artikelnummer	NUM	6 pos.
	omschrijving	ALFANUM	30 pos.
	aankoopprijs	NUM	4 pos.
	verkoopprijs	NUM	6 pos.
	winstperc	NUM	2 pos.

Dialog : tussen gebruiker en programma

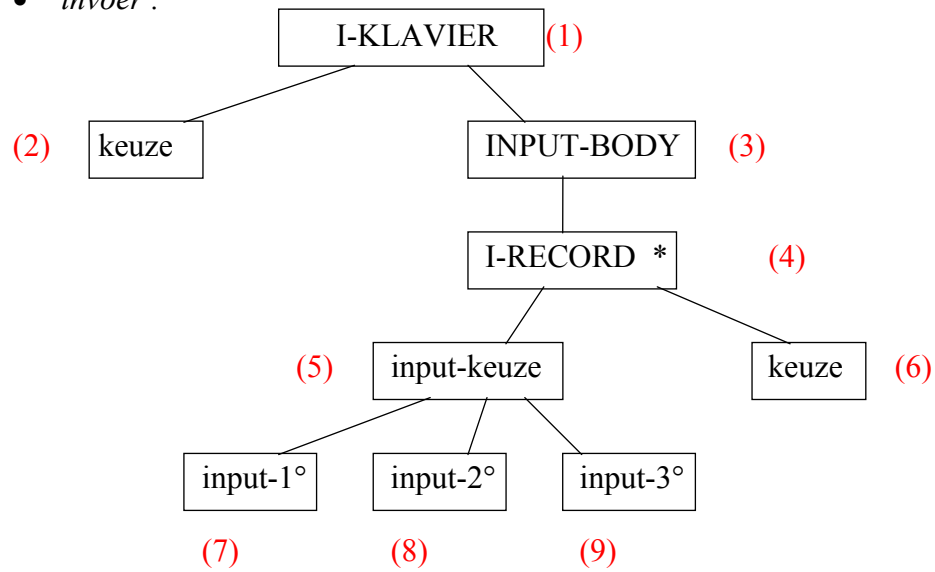


1) SND

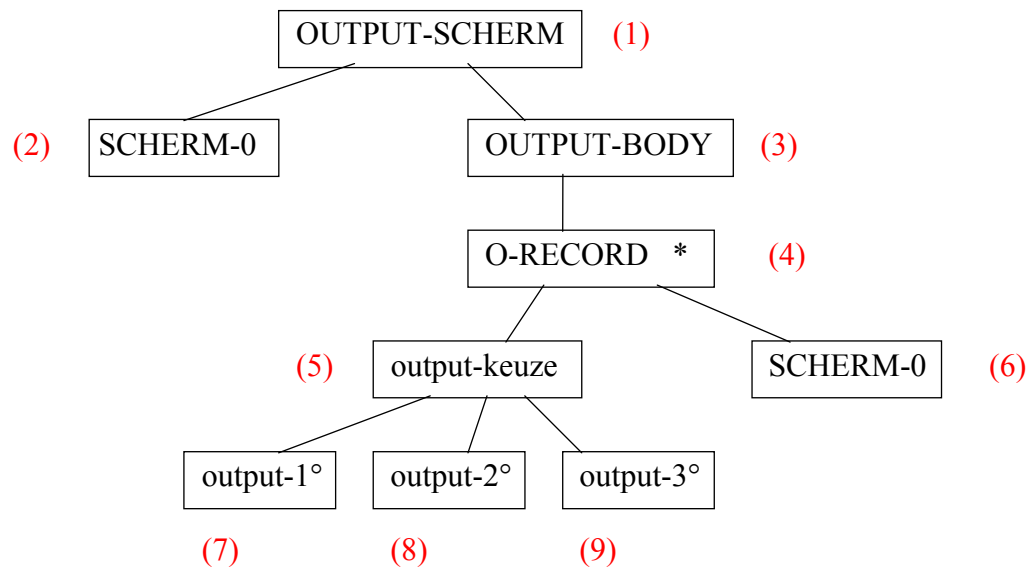


2) Gegevensstap - datastructuren

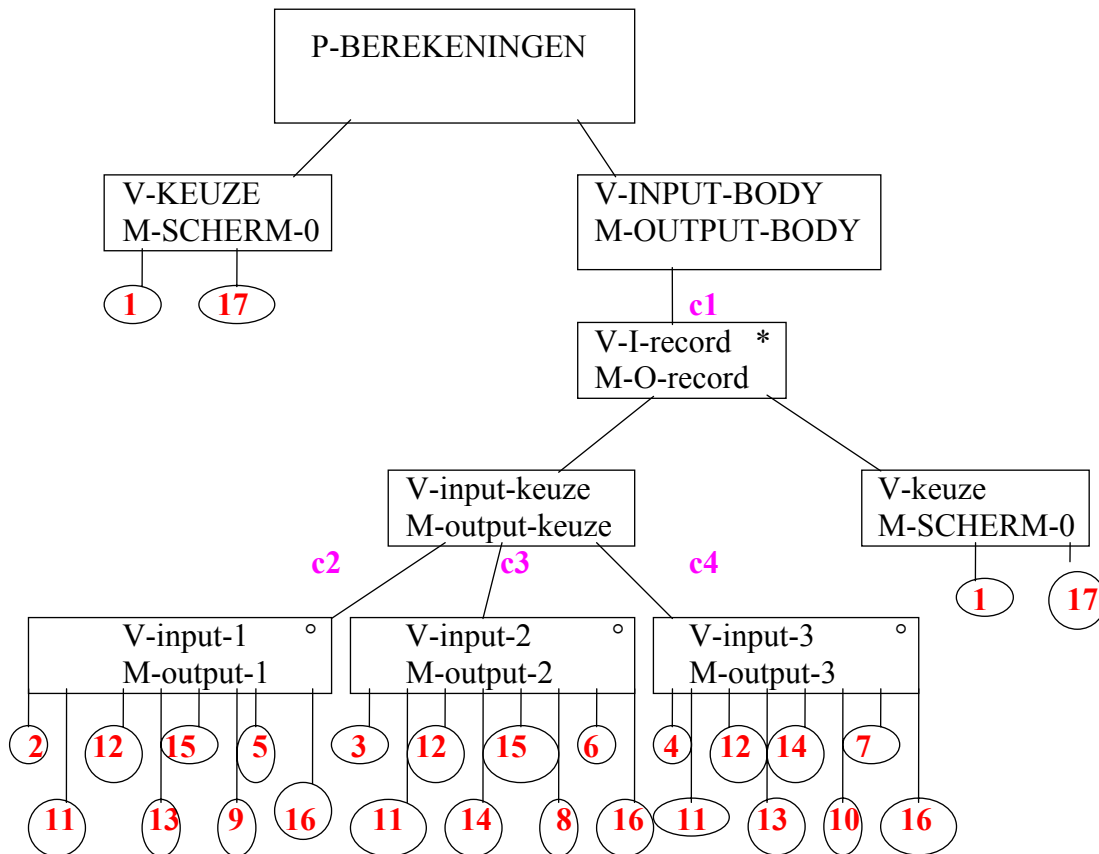
- *invoer* :



- *uitvoer* :



3) Programmastap



4) Opdrachten en condities

c1) ZOLANG (keuze \neq 0) DOE...

c2) ALS (keuze = 1) DAN...

c3) ALS (keuze = 2) DAN...

c4) ALS (keuze = 3) DAN...

1) voeruit (scherm, SCHERM-0)

2) voeruit (scherm, SCHERM-1)

3) voeruit (scherm, SCHERM-2)

4) voeruit (scherm, SCHERM-3)

5) voeruit (scherm, verkoopprijs)

6) voeruit (scherm, aankoopprijs)

7) voeruit (scherm, winstpercentage)

8) aankoopprijs \leftarrow verkoopprijs * (1 + winstpercentage / 100)

9) verkoopprijs \leftarrow aankoopprijs / (1 + winstpercentage / 100)

10) winstpercentage \leftarrow (verkoopprijs - aankoopprijs) * 100 / aankoopprijs

11) voerin (klavier, artikelnummer)

12) voerin (klavier, omschrijving)

13) voerin (klavier, aankoopprijs)

14) voerin (klavier, verkoopprijs)

15) voerin (klavier, winstpercentage)

16) voerin (klavier, toets)

17) voerin (klavier, keuze) + controle

5) Tekststap

P-Berekeningen-oef11

Begin

V-KEUZE-M-SCHERM-0

V-INPUT-BODY-M-OUTPUT-BODY

Einde

V-KEUZE-M-SCHERM-0

Begin

voeruit (scherm, SCHERM-0)

voerin (klavier, keuze)

Einde

V-INPUT-BODY-M-OUTPUT-BODY

Begin

ZOLANG (keuze ≠ 0)

DOE V-I-RECORD-M-O-RECORD

EINDE-ZOLANG-DOE

Einde

V-I-RECORD-M-O-RECORD

Begin

V-INPUT-KEUZE-M-OUTPUT-KEUZE

V-KEUZE-M-SCHERM-0

Einde

V-INPUT-KEUZE-M-OUTPUT-KEUZE

Begin

ALS (keuze = 1)

DAN V-INPUT-1-M-OUTPUT-1

EINDE-ALS-DAN

ALS (keuze = 2)

DAN V-INPUT-2-M-OUTPUT-2

EINDE-ALS-DAN

ALS (keuze = 3)

DAN V-INPUT-3-M-OUTPUT-3

EINDE-ALS-DAN

Einde

V-INPUT-1-M-OUTPUT-1

Begin

voeruit (scherm, SCHERM-1)

voerin (klavier, artikelnummer)

voerin (klavier, omschrijving)

voerin (scherm, aankoopprijs)

voerin (klavier, winstpercentage)

verkoopprijs ← aankoopprijs * (1 + winstpercentage / 100)

voeruit (scherm, verkoopprijs)

voerin (klavier, toets)

Einde

V-INPUT-2-M-OUTPUT-2

Begin

voeruit (scherm, SCHERM-2)
voerin (klavier, artikelnummer)
voerin (klavier, omschrijving)
voerin (klavier, verkoopprijs)
voerin (klavier, winstpercentage)
 $\text{aankoopprijs} \leftarrow \text{verkoopprijs} / (1 + \text{winstpercentage} / 100)$
voeruit (scherm, aankoopprijs)
voerin (klavier, toets)

Einde

V-INPUT-3-M-OUTPUT-3

Begin

voeruit (scherm, SCHERM-3)
voerin (klavier, artikelnummer)
voerin (klavier, omschrijving)
voerin (scherm, aankoopprijs)
voerin (klavier, verkoopprijs)
 $\text{winstpercentage} \leftarrow (\text{verkoopprijs} - \text{aankoopprijs}) * 100 / \text{aankoopprijs}$
voeruit (scherm, winstpercentage)
voerin (klavier, toets)

Einde

6) Implementatiestap

Cobolcode