

STARTPROGRAMMA

Citation for published version (APA):

Groot, W. J. (1974). *STARTPROGRAMMA*. (DCT rapporten; Vol. 1974.017). Technische Hogeschool Eindhoven.

Document status and date:

Gepubliceerd: 01/01/1974

Document Version:

Uitgevers PDF, ook bekend als Version of Record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

STARTPROGRAMMA

1. Inleiding
2. Programma beschrijving
3. Invoervolgorde
4. Foutenlijst
5. Opmerkingen

W. Groot
Dec.'74

1. INLEIDING

Er bestaan een aantal rekenprogramma's waarmee, gebaseerd op de elementenmethode, berekeningen aan constructies kunnen worden uitgevoerd. Dit betekent dat de constructie in elementen verdeeld moet worden, dus dat de elementenlocatie en de knooppuntscoördinaten bekend moeten zijn voordat zo'n programma kan rekenen.

Het is de taak van het STARTPROGRAMMA om een elementverdeling te realiseren en de gewenste gegevens in zodanige vorm uit te voeren dat deze via eenvoudige inleesprocedures door de rekenprogramma's kunnen worden opgenomen.

De voor een verdeling in aanmerking komende constructies zijn vlak of rotatorisch symmetrisch en de te genereren elementen zijn in het algemeen polygonen, waarvan naast de hoekpunten ook andere punten knooppunten kunnen zijn. Het merendeel der faciliteiten, zoals het gebruik van invoerbeperkende werkwijzen, heeft echter betrekking op het genereren van elementen met driehoekige vorm of doorsnede zoals TRIM3 of TRIAX6. Het is b.v. niet mogelijk om plotuitvoer te verwachten van een QUAM9-element, omdat dit element een knooppunt bezit dat niet op de omtrek ligt.

Het programma staat zowel in gecodeerde als ongecodeerde vorm op pack en levert uitvoer via de regeldrukker, en indien gewenst ook via de pons en de plotter.

2. PROGRAMMA-BESCHRIJVING

Met het startprogramma kan men een elementverdeling van een constructie realiseren, waarvan het resultaat wordt opgeslagen in de arrays LE, X en Y. Het array LE bevat informatie betreffende de elementlocatie, dus de knooppuntnummers van de elementen, terwijl de arrays X en Y resp. de x- en y-coördinaten van de knooppunten bevatten. Aan de hand van de operaties die op LE, X en Y worden uitgevoerd, zijn in het programma een aantal blokken te onderscheiden.

2.1. Vullen van LE, X en Y

Het programma bevat naast enkele procedures waarmee op conventionele manier de arrays gevuld kunnen worden (LEES01, LEES3) ook procedures die gebruikt kunnen worden indien een zekere regelmaat in de verdeling aanwezig is of aangebracht wordt (TOPOLOGY, MG01). Deze laatsten reduceren aanzienlijk de hoeveelheid in te voeren gegevens. Het is ook mogelijk om de uitvoer van het programma weer als deel van de invoer te gebruiken, om daar m.b.v. een procedure KOR kleine correcties in aan te brengen.

Als de arrays gevuld zijn tot een (op te geven) bovengrens, wordt met procedure VULTEST onderzocht of er zich gaten in bevinden en wordt tevens definitief het aantal knooppunten (AKK) en het aantal elementen (AEL) van de constructie vastgesteld. Een aantal van bovengenoemde procedures is attent op fouten in de eigen invoer en geeft bij constatering daarvan melding (zie o.a. bijgaande FOUTENLIJST).

Indien gewenst kan men hierna een element-transformatie laten uitvoeren m.b.v. procedure TRANSEL. Dit betekent dat voor driehoekige elementen knooppunten gegenereerd worden op de zijden, zonder dat daarbij de bandbreedte van de stijfheidsmatrix noemenswaard nadelig wordt beïnvloed t.o.v. een in dit opzicht ideale nummering. Er wordt een nieuw element locatie-array opgesteld (LEN) en opnieuw het totaal aantal knooppunten (AKKN) bepaald.

2.2. Operaties op LE, X en Y

Nu de constructie in elementen is verdeeld, kunnen er enkele operaties op de arrays worden uitgevoerd. Het doel hiervan is tweeledig, in de eerste plaats zijn er mogelijkheden om op enkele punten de juistheid van de elementverdeling te onderzoeken. Het is b.v. niet uitgesloten, dat door een onjuiste invoer een verdeling gegenereerd wordt, die niet aan de verwachting beantwoordt. In de tweede plaats zijn er mogelijkheden om de kwaliteit van de verdeling te beoordelen en te verbeteren. Onder kwaliteit wordt hier verstaan enerzijds hoe goed de vorm van een element lijkt op die van een ideaal-element, anderzijds verdient op numeriek-practische gronden een kleine (bruto) bandbreedte de voorkeur boven een grote. Operaties van de eerste soort zijn:

- a) onderzoek naar knooppunten, die dezelfde coördinaten bezitten (ELIM).
- b) onderzoek naar identieke elementen (IDEL 01).
- c) onderzoek naar elementen, waarbij één knooppunt meerdere malen voorkomt (PLATEL).

Het beoordelen van de vorm kan in dit programma alleen worden uitgevoerd voor driehoekige elementen met de procedure KAFAC.

Deze bepaalt een vormfactor gedefinieerd als $(\text{omtrek})^2 / \text{oppervlak}$, die voor een gelijkzijdige driehoek (ideaal-element) 20,8 bedraagt. Hoe groter de vormfactor, des te meer wijkt de vorm van het element af van die van het ideaal-element. Operaties van de laatste soort zijn b.v.:

- a) het repositioneren van de inwendige knooppunten, waardoor de vormfactor van de meeste elementen verbeterd wordt.
- b) het hernummeren van de gegenereerde knooppunten, met het doel om een kleinere (bruto) bandbreedte te verkrijgen.

Operatie b kan gewenst zijn indien, om het genereren te vereenvoudigen, een wilde knooppuntsnummering is ontstaan. Procedures om deze laatste operaties uit te voeren zijn in dit programma slechts in rudimentaire vorm aanwezig, d.w.z. dat deze procedures nog geen body bezitten. Indien, zoals meestal het geval zal zijn, voor een verdeling in driehoekige elementen de mesh-generator (MG 01) geactiveerd wordt, dan is het wel mogelijk om laatstgenoemde operatie a uit te voeren, echter alleen op door MG 01 gegenereerde elementen.

2.3. Uitvoer van LE(LEN), X en Y

Via de volgende randapparatuur wordt uitvoer geleverd:

2.3.1. Regeldrukker

Afhankelijk van de versie waarmee men werkt, verschijnt een programmalisting op de regeldrukker. Het verdient aanbeveling om de gecodeerde versie van het programma te gebruiken, omdat men in de programmalisting meestal niet geïnteresseerd zal zijn en bovendien vertaaltijd wordt uitgespaard (70 sec. CPU).

Tijdens de executie van het programma worden een aantal gegevens uitgevoerd, aan de hand waarvan de werking te controleren is.

De tussenuitvoer kan men vermeerderen door een in te voeren variabele "test" de waarde 1 te geven. Dit zal meestal slechts dan gebeuren, als onregelmatigheden in het programma geconstateerd of te verwachten zijn.

Indien het programma correct werkt, is slechts dat deel van de uitvoer interessant, dat voorafgegaan en afgesloten wordt door 3 rijen kruisjes. Hierin worden, voorafgegaan door een aantal gegevens die het probleem karakteriseren, de knooppuntscoördinaten X en Y en het array LE of LEN uitgevoerd. Als LEN wordt uitgevoerd betekent dit dat een element-transformatie heeft plaats gevonden, en worden door X en Y alleen de coördinaten van de hoekpunten van een element geleverd.

2.3.2. Ponser

Via de ponser kan kaartuitvoer geleverd worden van LE of LEN, X en Y. Het elementlocatie-array wordt voorafgegaan door 2 kaarten met tekst: de eerste kaart vermeldt het aantal elementen van de constructie, de tweede kaart het aantal knooppunten per element. Aan de knooppuntscoördinaten gaat één kaart met tekst vooraf, waarop het aantal knooppunten van de constructie vermeld wordt.

Als LEN wordt uitgevoerd geldt voor X en Y hetzelfde als bij 2.3.1.

Opgemerkt wordt dat, indien men de kaarten met tekst verwijdert, de ponsuitvoer van LE, X en Y opnieuw te gebruiken is als deel van de invoer voor het startprogramma. Dit kan voordeel bieden in het geval dat men een aantal correcties op een gegenereerde elementverdeling wil uitvoeren. Deze faciliteit geldt niet voor een uitvoer van LEN.

2.3.3. Plotter

De plotter levert een zeer waardevolle bijdragen in de beoordeling van een gegenereerde elementverdeling. In dit programma is daarom veel aandacht besteed om dit hulpmiddel optimaal te kunnen benutten. Van een elementverdeling of een te kiezen gedeelte daarvan, kan men plotfiguren laten uitvoeren op verschillend formaat en met verschillend strooisel, waarbij onder strooisel wordt verstaan het bijschrijven van knooppunten/of elementnummers. Voor de besturing van een en ander wordt verwezen naar de INVOERVOLGORDE.

3. INVOERVOLGORDE

De in te voeren gegevens zijn opgesplitst in drie delen, aangegeven met INVOER 3.1., INVOER 3.2. en INVOER 3.3. Met INVOER 3.1. wordt onderstaande (gebruikelijke) invoervolgorde bedoeld, terwijl de andere delen stukken deel invoer van INVOER 3.1. bevatten, die te uitgebreid zijn om daarin te worden opgenomen.

We merken op dat het programma in één aanbieding meerdere constructies in elementen kan verdelen. Als daarbij een elementverdeling mislukt t.g.v. fouten in de invoer die het programma kan constateren, dan kan toch de daaropvolgende elementverdeling correct worden uitgevoerd. Dit is mogelijk door de gegevens van elke constructie (ook de eerste) vooraf te laten gaan door het karakter #, dat door het programma als label gebruikt wordt.

INVOER 3.1.

ii		aantal verschillende constructies
per constructie	#	karakter dat aan de invoergegevens van een constructie vooraf gaat.
	test	test=1: tijdens de executie van het programma wordt extra uitvoer gegeven. test 1: geen extra uitvoer test= 12345: de normale ponsuitvoer wordt onderdrukt.
	ael	aantal elementen van de constructie of een daarvoor geschatte bovengrens.
	ake	aantal knooppunten per element. Als de elementen TRIM6 of TRIAX6 gebruikt worden, moet ake=6 zijn; voor de rest van de invoer kunnen deze elementen als TRIM3 of TRIAX3 behandeld worden.
	ac	aantal coördinaten per knooppunt.
	akk	aantal knooppunten van de constructie of een daarvoor geschatte bovengrens.
	tasn	totaal aantal subnetten bij gebruik van de procedure MG01. Als MG01 niet gebruikt wordt, dantasn=1

per constructie	almax	het maximum aantal lijnen van een subnet bij gebruik van de procedure MG01 Als MG01 niet gebruikt wordt, dan almax=1.
	art	aantal regels tekst waaruit de programma-identificatie bestaat; $art \leq 3$.
	'('regel 1')'	art regels tekst. Een regel tekst kan 60 karakters bevatten, terwijl bij minder karakters geadviseerd wordt om de resterende posities, i.v.m. de leesbaarheid, aan te vullen met spaties.
	'('regel art')'	Elke regel wordt voorafgegaan door de karakters '(' en afgesloten door ')'
	n	aantal manieren waarop geometrische en topologische gegevens worden ingelezen; $n \leq 10$.
	j_1, j_2, \dots, j_n	met j_i wordt de manier aangegeven waarop geometrische en/of topologische gegevens worden ingelezen. Het is niet noodzakelijk dat: $m > n \Rightarrow j_m > j_n$ m.a.w. de volgorde waarin de inleesmogelijkheden worden gebruikt is niet van belang $j_i = 1$: gebruik van procedure LEES 01 $j_i = 3$: " LEES 3 $j_i = 4$: " TOPOLOGY $j_i = 5$: " MG01 $j_i = 6$: " KOR
	INVOER 3.2	invoergegevens voor de manieren j_1, j_2, \dots, j_n (zie aldaar).
	m	aantal operaties die op de ingelezen geometrische en topologische gegevens moeten worden uitgevoerd.
	k_1, k_2, \dots, k_m	aanduiding van de gewenste operaties. $k_i = 1$: gebruik van procedure ELIM $k_i = 2$: " IDEL 01 $k_i = 3$: " PLATEL $k_i = 4$: " KAFAC $k_i = 5$: " HUSSEL $k_i = 6$: " RENUM 01
		De laatste twee procedures zijn nog niet operationeel.

per constructie per figuur	afig	aantal gewenste figuren.
	n1	<p>hiermee geeft men de elementen aan die getekend moeten worden.</p> <p>n1=1 : zelfde elementen als in het voorgaande plaatje</p> <p>n1=2 : alle elementen</p> <p>n1=3 : een deel van de elementen, nl. de elementen die zich bevinden in het door rechthoek1 beschreven gebied.</p> <p>n1=-3: een deel van de elementen, nl. de elementen die zich geheel bevinden buiten het door rechthoek1 beschreven gebied.</p>
	rechthoek1	<p>Deze invoer bestaat uit 4 getallen die een rechthoek karakteriseren, nl.: xmin, ymin, xmax, ymax.</p> <p>Voorwaarde voor een goede werking is dat $xmin < xmax$ en $ymin < ymax$ gekozen wordt.</p> <p>rechthoek1 wordt alleen gegeven als $abs(n1)=3$.</p>
	form	<p>aanduiding voor het gewenste formaat van het plaatje.</p> <p>form=1 : maximaal formaat 270x810</p> <p>form=2 : "A3-formaat" 270x400</p> <p>form=3 : "A4-formaat" 200x270</p> <p>We merken hierbij op dat de genormaliseerde afmetingen van de formaten A3 en A4 resp. zijn: 297 x 420 en 210 x 297.</p>
n2	<p>aanduiding voor de elementen of knooppunten waar strooisel gewenst wordt.</p> <p>n2=1 : alleen randknooppunten</p> <p>n2=2 : alle elementen</p> <p>n2=-2: geen elementen</p> <p>n2=3 : een deel van de elementen, nl. de elementen die zich geheel bevinden in het door rechthoek2 beschreven gebied.</p> <p>n2=-3: een deel van de elementen, nl. de elementen die zich geheel bevinden buiten het door rechthoek2 beschreven gebied.</p>	
rechthoek2	<p>Deze invoer bestaat uit 4 getallen die een rechthoek karakteriseren, nl.: xmin, ymin, xmax, ymax.</p> <p>Ook hier is nodig dat $xmin < xmax$ en $ymin < ymax$.</p> <p>rechthoek1 wordt alleen gegeven als $abs(n2)=3$</p>	

	s	aanduiding voor het gewenste type strooisel. s=1 : bij schrijven van knooppuntnummers s=2 : bij schrijven van elementnummers s=3 : bij schrijven van knooppunt- en elementnummers s mag niet worden gegeven als n2=-2.
--	---	--

INVOER 3.2.

Onderstaand volgen de in te voeren gegevens, die nodig zijn bij het gebruik van de verschillende inleesprocedures.

LEES 01:

Met deze procedure worden knooppuntscoördinaten ingelezen.

Invoervolgorde:

	k1	x(k1)	y(k1)	z(k1)	nummer van een knooppunt, gevolgd door de coördinaten van dat knooppunt. Voor tweedimensionale elementen wordt z(ki) niet ingevuld.	
	kn	x(kn)	y(kn)	z(kn)		
0					Afsluiting van de gegevens van deze procedure.	

LEES 3:

Met deze procedure wordt de elementlocatie ingelezen.

Invoervolgorde:

	e11	k1(e11)	kake(e11)	nummer van een element, gevolgd door de knooppunten van dat element.	
	eln	k1(eln)	kake(eln)		
0				Afsluiting van de gegevens van deze procedure.	

TOPOLOGY:

Deze procedure genereert de elementlocatie. Voor de werking van deze procedure wordt verwezen naar de diverse aanwezige beschrijvingen.

Invoervolgorde:

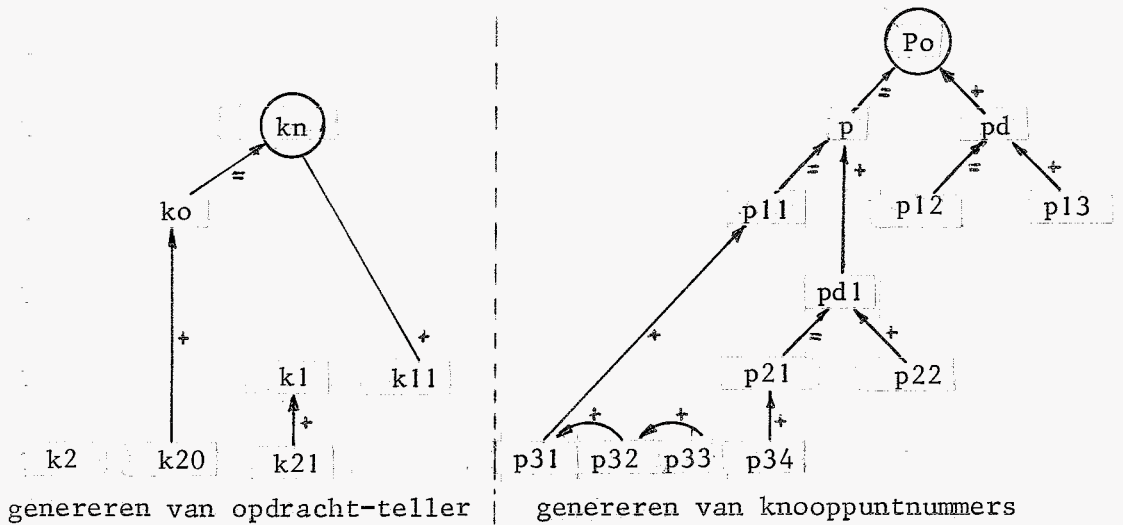
	eel	nummer van het eerstvolgende element.	
	at		aantal opdrachten
at maal	i		
	invoer i	invoergegevens voor opdrachttype i.	

invoer-1 : ko (p11,p12,p13)⁽¹⁾ - - (p11,p12,p13)^(ake)

invoer-2 : invoer-1
 k1,k11 (p21,p22)⁽¹⁾ - - (p21,p22)^(ake)

invoer-3 : invoer-2
 k2,k20,k21 (P31,p32,p33,p34)⁽¹⁾ - - (p31,p32,p33,p34)^(ake)

Schematische weergave van de werking van procedure TOPOLOGY:

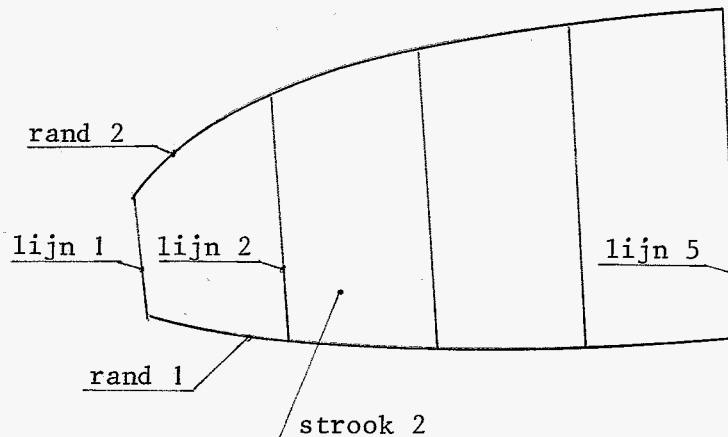


genereren van opdracht-teller

genereren van knooppuntnummers

MG 01:

Met deze procedure kunnen de knooppuntscoördinaten en de element-locatie worden gegenereerd; verdeling van de constructie in subnetten is mogelijk. Ter verduidelijking van enkele in het vervolg te gebruiken termen volgt hier een schets van een subnet.



De invoervolgorde is voorzien van de nummers 1 t/m 14 om de communicatie te vergemakkelijken. In de FOUTENLIJST wordt b.v. naar deze nummering verwezen.

Invoervolgorde:

eel	nummer van het eerstvolgende element
esn	nummer van het eerstvolgende subnet
asn	aantal te genereren subnetten.

1. Per subnet worden de invoergegevens (1) ingevoerd:

a1	aantal lijnen
k1,a1,b1,c1	teller + groepje van 3 topologische variabelen t.b.v.
	het genereren van de aantallen intervallen van de
	lijnen
kn,an,bn,cn	afsluiting van de intervalgegevens
0	lijnummer + nummer van het eerste knooppunt waarvoor
l1,n1	men dat wenst op te geven.
ln,nn	afsluiting van lijnummergegevens.
0	

Invoergegevens t.b.v. het koppelen van subnetten (2 en 3)

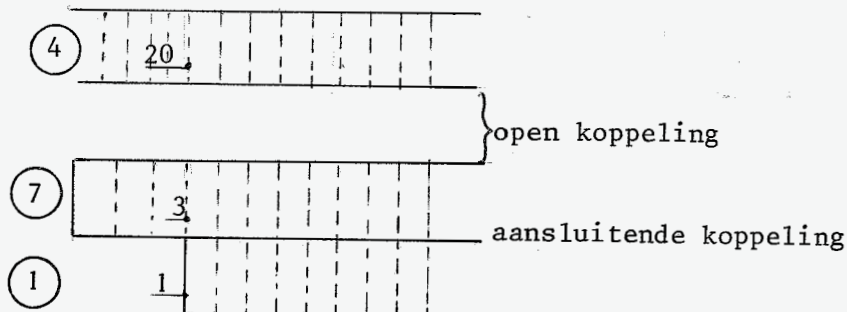
Subnetten kunnen op twee manieren gekoppeld worden nl. langs randen en langs lijnen; de te geven voorbeelden zullen dit kunnen verduidelijken. De koppelingsoperaties kan men willekeurig vaak na elkaar toepassen, mits de volgorde logisch is. De totale koppelingsinvoer wordt afgesloten door het getal 0.

2. Koppeling langs randen

koptyp	Met kptyp=1 geven we koppeling langs een rand aan.
atkn	aantal te koppelen subnetten.
nnr(1), ac(1)	nummers van de te koppelen subnetten in volgorde en
	gescheiden door de koppelingscode ac.
nnr(atkn-1),ac(atkn-)	ac=1: aansluitende koppeling
nnr (atkn)	ac=0: open koppeling

akl | aantal te koppelen lijnen
 l1(nnr(1)) | nummers van de eerste te koppelen lijn van elk subnet.
 |
 l1(nnr(atkn))

Voorbeeld:



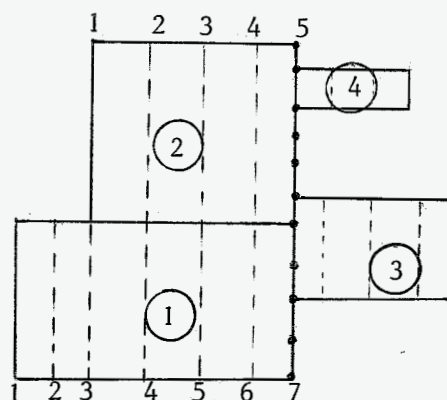
De netten 1,7 en 4 worden gekoppeld met de volgende invoer:

1,3, 1, 1, 7, 0, 4, 9, 1, 3, 20.

3. Koppeling langs lijnen

koptyp | Met koptyp=2 geven we koppeling langs een lijn aan.
 atkn | aantal te koppelen subnetten
 apkl | totaal aantal punten op de lijn waarlangs gekoppeld wordt
 n(1), l(1), p(1) | netnummer, lijnnummer en positie van het 1e punt van
 | | | de lijn op de koppelingslijn (atkn maal).
 n(atkn), l(atkn), p(atkn)

Voorbeeld:



De netten 1,2,3 en 4 worden gekoppeld met de volgende invoer:

2,4,11, 1,7,1, 2,5,5, 3,1,3, 4,1,9,

0 | afsluiting van de koppelingsinvoer.

Invoergegevens per subnet (4 t/m 14)

4.
autverh

In de punten 10 t/m 14 wordt aangegeven hoe en welke lijnen in intervallen verdeeld moeten worden. Lijnen die daarin niet worden gespecificeerd kunnen door de keuze van "autverh" als volgt worden verdeeld:

autverh=0: verdeling in gelijke intervallen. (zie 10)

autverh=2: verdeling in ongelijke intervallen aan de hand van de afstanden van de randpunten. (zie 11)

diag

De keuze van de diagonalen in een strook kan men sturen met diag. Meestal zal de kortste diagonaal in een rechthoek de voorkeur verdienen, maar er kunnen zich situaties voordoen waarbij de keuze van de kortste diagonaal leidt tot een bepaald lijnenpatroon dat mogelijk niet gewenst is. Binnen bepaalde toleranties is het dan mogelijk om de langste diagonaal te kiezen.

diag=0 : de diagonalen in de volgende strook worden zoveel mogelijk tegengesteld genomen aan die van de huidige strook.

diag=1 : de diagonalen in de volgende strook worden zoveel mogelijk gelijk genomen aan die van de huidige strook.

diag=2 : de kortste diagonaal wordt per strook gekozen.

amr

aantal malen repositioneren van de inwendige knooppunten.

Invoer voor de randen van een subnet (5 t/m 8)

We voeren achtereenvolgens rand 1 en rand 2 in. Een rand wordt ingevoerd door het eerste punt van die rand te geven (5), vervolgens naar keuze een aantal malen een greep te doen uit de opdrachten 5, 6 of 7 en tenslotte deze invoer af te sluiten met het getal 0. (8)

5.

keuze		keuze=1; we geven een enkel punt aan.
x1,y1		coördinaten van het punt.

6.

keuze		keuze=2; we geven een reeks punten op een recht lijnstuk aan
aint		het aantal intervallen op het lijnstuk
vf		vf=0 : verdeling in gelijke intervallen vf=1 : verdeling in ongelijke intervallen
xe,ye		coördinaten van het eindpunt van het lijnstuk
vh(1),---vh(aint)		verhoudingsgetallen voor de verdeling in intervallen. De verhoudingsgetallen alleen opgeven als vf=1.

7.

keuze		keuze=3; we geven een reeks punten op een cirkelboog aan.
aint		het aantal intervallen op de cirkelboog
vf		vf=0 : verdeling in gelijke intervallen vf=1 : verdeling in ongelijke intervallen
xt,yt		coördinaten van een tussen begin- en eindpunt gelegen punt van de cirkelboog.
xe,ye		coördinaten van het eindpunt van de cirkelboog.
vh(1),---vh(aint)		verhoudingsgetallen voor de verdeling in intervallen. De verhoudingsgetallen alleen opgeven als vf=1

8.

0		Afsluiting van invoergegevens van een rand.
---	--	---

Typering en verdeling van lijnen in intervallen (9 t/m 14)

In (1) wordt het aantal intervallen per lijn opgegeven. In het nu volgende wordt de intervallengte nader bepaald. Ook is het mogelijk om een lijn te typeren als cirkelvormig of samengesteld.

9.

k1, x1, y1		We geven per lijn het nummer en de coördinaten van een tussen begin- en eindpunt gelegen punt van de cirkelboog.
kn, xn, yn		
0		Afsluiting van de gegevens voor cirkelvormige lijnen.

10. Lijnen die verdeeld worden in gelijke intervallen
- | | |
|---|--|
| k_1, a_1, b_1, c_1
$\vdots \quad \vdots \quad \vdots \quad \vdots$
kn, a_n, b_n, c_n

0 | <p>We genereren de lijnnummers m.b.v. topologische variabelen.</p> <p>Voorbeeld: $\{k, a, b, c, \} = \{4, 2, 1, 1\}$ genereert 4 knooppunten nl. 2, 3, 5 en 8.</p> <p>Afsluiting van deze gegevens.</p> |
|---|--|
11. Lijnen die verdeeld worden in ongelijke intervallen (a)
- | | |
|---|---|
| k_1, a_1, b_1, c_1
$\vdots \quad \vdots \quad \vdots \quad \vdots$
kn, a_n, b_n, c_n

0 | <p>De intervallenlengte wordt bepaald aan de hand van de afstanden van de randpunten. We geven de lijnnummers m.b.v. een set topologische variabelen (als in 10)</p> <p>afsluiting van deze gegevens.</p> |
|---|---|
12. Lijnen die verdeeld worden in ongelijke intervallen (b)
- | | |
|---|---|
| $k_1, vh(1,1), \dots, vh(1, aint(k_1))$
\vdots
$kn, vh(n,1), \dots, vh(n, aint(kn))$

0 | <p>De intervallenlengte wordt bepaald aan de hand van verhoudingsgetallen. We geven per lijn het lijnnummer en zoveel verhoudingsgetallen als er intervallen zijn (zie ook 1).</p> <p>Afsluiting van deze gegevens.</p> |
|---|---|
13. Lijnen die verdeeld worden in ongelijke intervallen (c)
- | | |
|---|--|
| $k_1, vh11, vh12, vh13$
$\vdots \quad \vdots \quad \vdots \quad \vdots$
$kn, bh11, vhn2, vhn3$

0 | <p>De intervallenlengte wordt bepaald aan de hand van 3 verhoudingsgetallen. We geven per lijn het lijnnummer en een verhoudingsgetal voor het begin, midden en eind van de lijn.</p> <p>afsluiting van deze gegevens.</p> |
|---|--|
14. Samengestelde lijnen
- | | |
|---|--|
| $k_1, \text{randinvoer } 1$
$\vdots \quad \vdots$
$kn, \text{randinvoer } n$

0 | <p>De intervallenlengte wordt bepaald door zelf punten op de lijn aan te geven. We geven per lijn het lijnnummer en de invoer die ook voor een rand gebruikt wordt (zie 5 t/m 8).</p> <p>afsluiting van deze gegevens.</p> |
|---|--|
- De invoer 4 t/m 14 herhalen voor het volgende subnet.
- | | |
|------------|--|
| 12345 6789 | afsluiting van de invoergegevens van procedure MG01. |
|------------|--|

KOR :

Deze procedure biedt de mogelijkheid om de elementverdeling van een constructie te corrigeren door knooppunten resp. elementen toe te voegen of te verwijderen.

Invoervolgorde:

k_1, k_2, \dots, k_n

-1
 $noud^{(1)}, a^{(1)}, x_1^{(1)}, y^{(1)}$
 | |
 $x_a^{(1)}, y_a^{(1)}$

|
 |
 |
 |
 |
 |
 |
 |
 |
 |
 $noud^{(k)}, a^{(k)}, x_1^{(k)}, y_1^{(k)}$
 | |
 $x_a^{(k)}, y_a^{(k)}$

-1
 k_1, k_2, \dots, k_n

-1
 $noud^{(1)}, a^{(1)}, n(1,1)^{(1)}, \dots, n(ake,1)^{(1)}$
 | |
 $n(1,a)^{(1)}, \dots, n(ake,a)^{(1)}$

|
 |
 |
 |
 |
 |
 |
 |
 |
 |
 $noud^{(k)}, a^{(k)}, n(1,1)^{(k)}, \dots, n(ake,1)^{(k)}$
 | |
 $n(1,a)^{(k)}, \dots, n(ake,a)^{(k)}$

-1

De nummers van de te verwijderen knooppunten; de volgorde wordt bepaald door $k_i < k_{i+1}$

Afsluiting van te verwijderen knooppunten.

Na noud worden a knooppunten toegevoegd, gevolgd door (in volgorde van toevoegen) hun coördinaten.

N.B. alleen t.b.v. de elementlocatie worden de toegevoegde knooppunten in de procedure negatief genummerd, te beginnen bij -1.

De volgorde van $noud^{(i)}$ wordt bepaald door de conditie $noud^{(i)} < noud^{(i+1)}$.

afsluiting van toe te voegen knooppunten.

de nummers van de te verwijderen elementen;

de volgorde wordt bepaald door $k_i < k_{i+1}$

afsluiting van te verwijderen elementen.

Na noud worden a elementen toegevoegd, gevolgd door (in volgorde van toevoegen) hun knooppunten;

nieuwe knooppunten hebben een negatief nummer.

De volgorde van $noud^{(i)}$ wordt bepaald door de conditie $noud^{(i)} < noud^{(i+1)}$

afsluiting van toe te voegen elementen.

INVOER 3.3.

Onderstaand volgen de in te voeren gegevens die nodig zijn voor het uitvoeren van de verschillende operaties op de gegenereerde verdeling.

ELIM:

Deze procedure onderzoekt of er verschillende knooppuntnummers zijn met dezelfde coördinaten. Als dit het geval blijkt te zijn, volgt een melding op de regeldrukker. Bij een in te voeren grootheid eps zijn de knooppunten i en j identiek als geldt:

$$(|x_i - x_j| \leq \text{eps}) \wedge (|y_i - y_j| \leq \text{eps}) \wedge (|z_i - z_j| \leq \text{eps}).$$

Invoervolgorde:

eps	toegestane absolute afwijking van de coördinaten van een knooppunt
-----	--

IDEL 01:

Onderzoek naar identieke elementen.

Invoervolgorde:

n	aanduiding van de groep aaneengesloten elementen, die worden onderzocht.
	n=1: de elementen ℓ_1 t/m ℓ_2 .
	n≠1: de elementen 1 t/m ael
ℓ_1, ℓ_2	deze getallen alleen geven als n=1

PLATEL:

Deze procedure onderzoekt elementen op het dubbel voorkomen van knooppunten.

Invoervolgorde:

n	aanduiding van de groep aaneengesloten elementen die worden onderzocht.
	n=1: de elementen ℓ_1 t/m ℓ_2
	n≠2: de elementen 1 t/m ael.
ℓ_1, ℓ_2	deze getallen alleen geven als n=1.

KAFAC:

De procedure bepaalt een vormfactor voor-momenteel alleen driehoekige-elementen. De vormfactor wordt hier gedefinieerd als $(\text{omtrekt})^2 / \text{oppervlak}$.

4. FOUTENLIJST

Bij gebruik van de procedures waarmee de geometrische en topologische gegevens worden ingelezen, kunnen fouten gemaakt worden als het vergeten van een getal of het invoeren van een getal dat zekere aan dat getal te stellen grenzen overschrijdt, waarbij de laatste soort fout vaak een gevolg is van de eerste. Een aantal van deze fouten kan door de betreffende procedure zelf geconstateerd worden en resulteert dan in een gecodeerde foutmelding. Het programma voert in dat geval de inhoud van array top uit, soms nog voorafgegaan door een globale aanduiding van de plaats waar de fout is gemaakt. Het array top is gedeclareerd als integer array top [0:10,1:3], en wordt tevoren op nul geïnitieerd. De eerste kolom van top bevat gegevens over het aantal en type manieren waarop geometrische en topologische gegevens worden ingelezen.

top [0,1] = aantal manieren=n (zie INVOERVOLGORDE)
 top [1,1] t/m top [1,n] = te gebruiken procedures=j1,j2,---,jn (zie invoer-
 volgorde)

In de 2e en 3e kolom wordt normaal bijgehouden hoeveel elementen resp. hoeveel knooppunten er door het gebruik van de betreffende procedure zijn bijgekomen, waarbij opgemerkt wordt dat bij het gebruik van MG 01 de knooppunten, die tot 2 netten behoren, ook 2 maal geteld worden. Echter als de procedure een fout signaleert worden in de kolommen 2 en 3 van top gegevens over dit fout geleverd. Als het array top tot de ii-e rij gevuld is voeren we de volgende afkortingen in:

top [0,2] =: foutnummer
 top [0,3] =: A
 top [ii,2] =: B
 top [ii,3] =: C

Voor de diverse inleesprocedures volgt nu een foutenlijst, waarbij voor iedere fout de relevante gegevens van A, B en C worden geleverd; het nummer van de fout stemt overeen met het reeds eerder genoemde foutnummer.

LEES 01

1. Er wordt een niet bestaand knooppuntnummer ingelezen.

A= het niet bestaande knooppuntnummer.

C= het aantal ingelezen knooppunten.

TOPOLOGY

1. De variabele type heeft niet de juiste waarde
 A= waarde van type
 B= aantal gegenereerde elementen
2. De arraygrens van le wordt overschreden
 B= aantal gegenereerde elementen
3. Een te groot knooppuntnummer wordt gegenereerd (>akk)
 A= knooppuntnummer
 B= aantal gegenereerde elementen

LEES 3

1. Een niet bestaand elementnummer wordt ingelezen
 A= elementnummer
 B= aantal ingelezen elementen
2. Een niet bestaand knooppuntnummer wordt ingelezen
 A= knooppuntnummer
 B= aantal goed ingelezen elementen

MG 01

De invoer voor de procedure MG 01 wordt in het programma gelezen en opgeslagen in het één-dimensionale array IIP1. Tegelijkertijd wordt deze invoer via de regeldrukker geprint om het opzoeken van eventuele fouten te vereenvoudigen. Bij aanroep van MG 01 wordt IIP1 als actuele parameter meegegeven. Na afloop van de procedure wordt het aantal door de procedure verwerkte getallen uitgevoerd, alsmede de waarde van het laatst verwerkte getal.

Een groot aantal gedeelten van de invoergegevens wordt afgesloten door het getal 0. Het is moeilijk te testen of deze nullen correct zijn ingevoerd. Aangeraden wordt daarom, om bij moeilijk verklaarbare fouten na te gaan, of de afsluitende nullen voor en na het gedeelte van de invoer dat in behandeling is, correct zijn geplaatst.

1. Het aantal lijnen van een subnet is groter of kleiner dan toegestaan;
 $a1 < 0$ of $a1 > almax$.
 B=almax
 C=a1
 Zie invoer deel 1 en vergelijk dit met almax uit INVOER 3.1

2. Bij het genereren van de aantallen intervallen per lijn is het aantal lijnnummers kleiner dan het aantal lijnen in het betreffende subnet.

B= aantal lijnen

C= aantal gegenereerde lijnnummers

Zie invoer deel 1.

3. Bij het genereren van het aantal intervallen van de lijnen treedt voor de topologische variabelen een teller $k < 0$ op.

A=k

Zie invoer deel 1.

4. Bij het genereren van de aantallen intervallen per lijn is het aantal lijnnummers groter dan het aantal lijnen in het betreffende subnet.

B= aantal lijnen.

C: er worden minstens C lijnnummers gegenereerd.

Zie invoer deel 1.

5. Het kleinste aantal intervallen dat wordt gegenereerd, $\text{aintmin} < 0$

A= aintmin

Zie invoer deel 1.

6. Bij de lijnen waarvan het eerste knooppunt wordt opgegeven, is een opgegeven lijnummer $\ell < 0$ of $\ell >$ aantal lijnen (a_ℓ) in het betreffende subnet.

B= a_ℓ

C= ℓ

Zie invoer deel 1.

7. Bij de lijnen waarvan het eerste knooppunt wordt opgegeven is zo'n eerste knooppunt $k < 0$ of $k >$ akk.

B= akk

C= k

Zie invoer deel 1.

8. Bij het koppelen van de netten krijgt de variabele koptyp een andere waarde dan 0,1,2, of 3.

A= koptyp

Zie invoer deel 2 of 3.

9. Bij het koppelen van de netten is het opgegeven aantal te koppelen netten $\text{atkn} < 1$ of $\text{atkn} >$ asn (=aantal subnetten bij deze aanroep van MG 01)

B= asn

C= atkn

Zie invoer deel 2 of 3.

10. Bij het langs een rand koppelen van netten is het eerste opgegeven netnummer $\langle \text{esn (1e subnet) of netnummer} \rangle \text{esn} + \text{asn} - 1$ (= laatste netnummer).
- A= netnummer
 B= esn
 C= asn
 Zie invoer deel 2.
11. Bij het langs een rand koppelen van subnetten is de opgegeven aansluitcode $\text{ac} \neq 0$ en $\text{ac} \neq 1$.
- A=ac
 Zie invoer deel 2.
12. Bij het langs een rand koppelen van netten is een opgegeven netnummer (niet het eerste) $\langle \text{esn of netnummer} \rangle \text{esn} + \text{asn} - 1$.
- A= netnummer
 B= esn
 C= asn
 Zie invoer deel 2.
13. Bij het langs een rand koppelen van netten is het opgegeven aantal lijnen die gekoppeld worden $\text{ak} \ell < 1$ of $\text{ak} \ell > \text{aantal lijnen in een van de te koppelen netten}$.
- B= aantal lijnen
 C= $\text{ak} \ell$
 Zie invoer deel 2
14. Bij het langs een rand koppelen van netten is een opgegeven eerste lijnnummer $\ell 1 < 1$ of $\ell 1 > \text{aantal lijnen in het betreffende subnet}$.
- B= aantal lijnen
 C= $\ell 1$
 Zie invoer deel 2.
15. Bij het langs een lijn koppelen van netten is het totale aantal punten op de lijn waarlangs gekoppeld wordt $\text{ap} \ell < 1$
- A= $\text{ap} \ell$
 Zie invoer deel 3.
16. Bij het langs een lijn koppelen van netten valt een opgegeven netnummer buiten de range: $\text{esn} < \text{netnummer} < \text{esn} + \text{asn} - 1$.
- A= netnummer
 B= esn
 C= asn
 Zie invoer deel 3.

17. Bij het langs een lijn koppelen van netten valt een opgegeven lijnnummer buiten de range: $1 \leq \text{lijnnummer} \leq \text{aantal lijnen}$ in het betreffende net.
 B= aantal lijnen
 C= lijnnummer
 Zie invoer deel 3.
18. Bij het langs een lijn koppelen van netten valt een opgegeven positie(pos) van het eerste knooppunt van een lijn buiten de range: $1 \leq \text{pos} \leq \text{apkl}$ (=totaal aantal punten op de koppellijn).
 B= apkl
 C= pos.
 Zie invoer deel 3.
19. De opgegeven waarde van autverh is niet gelijk aan 0,1,2 of 3.
 A= autverh.
 Zie invoer deel 4.
20. De opgegeven waarde van diag is ongelijk aan 0,1 of 2.
 A= diag.
 Zie invoer deel 4.
21. Bij het genereren van een samengestelde lijn (event.een rand) is de waarde van keuze ongelijk aan 0,1,2 of 3.
 A= keuze
 Zie invoer deel 5 t/m 8 of deel 14.
22. Voor een samengestelde lijn (event. rand) worden te veel punten gegenereerd.
 B= aantal te genereren punten
 C= aantal gegenereerde punten
 Zie invoer deel 5 t/m 8 of deel 14.
23. Voor een samengestelde lijn (event. rand) worden te veel punten gegenereerd.
 B= aantal te genereren punten
 C: aantal gegenereerde punten $\geq C$
 Zie invoer deel 5 t/m 8 of deel 14.
24. Bij het genereren van een recht of cirkelvormig lijnstuk van een samengestelde lijn is het opgegeven aantal intervallen < 1 .
 A= aantal intervallen
 Zie invoer deel 6,7 of 14.

25. Bij het genereren van een recht of cirkelvormig lijnstuk van een samengestelde lijn (event. rand) is de waarde van de variabele $vf \neq 0$ of $vf \neq 1$.
 $A = vf$
 Zie invoer deel 6,7 of 14
26. De invoer voor een samengestelde lijn begint niet met keuze=1.
 $A = \text{keuze}$
 Zie invoer deel 5 of 14.
27. Bij het genereren van een recht of cirkelvormig lijnstuk van een samengestelde lijn (event. rand) is de waarde van een opgegeven verhoudingsgetal $vh \leq 0$.
 $A = vh$
 Zie invoer deel 6,7 of 14.
28. Het lijnnummer van een cirkelvormige lijn valt buiten de range:
 $0 \leq \text{lijnnummer} \leq \text{aantal lijnen in het betreffende net.}$
 $B = \text{aantal lijnen.}$
 $C = \text{lijnnummer.}$
 Zie invoer deel 9.
29. Bij het genereren van de nummers van de lijnen die in gelijke intervallen verdeeld worden is voor een topologische variabele de teller $k < 0$.
 $A = k$
 Zie invoer deel 10.
30. Voor de lijnen die in gelijke intervallen verdeeld worden, worden er meer nummers gegenereerd dan er lijnen zijn in het betreffende net.
 $B = \text{aantal lijnen}$
 $C: \text{aantal gegenereerde lijnnummers} \geq C.$
31. Het hoogste nummer van de lijnen die in gelijke intervallen verdeeld worden, is hoger dan het aantal lijnen in het betreffende net.
 $B = \text{aantal lijnen.}$
 $C = \text{hoogste lijnnummer}$
 Zie invoer deel 10.
32. Het laagste nummer van de lijnen die in gelijke intervallen verdeeld worden, $\&min < 1$.
 $A = \&min$
 Zie invoer deel 10

33. Bij het genereren van de nummers van de lijnen die aan de hand van de afstand van de randpunten verdeeld worden, is voor een topologische variabele de teller $k < 0$.
 $A = k$
 Zie invoer deel 11.
34. Voor de lijnen die aan de hand van de afstanden van de randpunten verdeeld worden, worden er meer nummers gegenereerd dan er lijnen zijn in het betreffende subnet.
 $B =$ aantal lijnen
 $C:$ aantal gegenereerde lijnummers $\geq C$.
 Zie invoer deel 11.
35. Het hoogste nummer van de lijnen die aan de hand van de afstanden van de randpunten in ongelijke intervallen worden verdeeld, is hoger dan het aantal lijnen in het betreffende subnet.
 $B =$ aantal lijnen
 $C =$ hoogste lijnummer
 Zie invoer deel 11.
36. Het laagste nummer van de lijnen die aan de hand van de afstanden van de randpunten verdeeld worden, $l_{\min} < 1$.
 $A = l_{\min}$
 Zie invoer deel 11.
37. Een nummer van een lijn met opgegeven verhoudingsgetallen valt buiten de range: $0 \leq \text{lijnummer} \leq \text{aantal lijnen}$ in het betreffende net.
 $B =$ aantal lijnen
 $C =$ lijnummer
 Zie invoer deel 12.
38. Een opgegeven verhoudingsgetal voor het verdelen van een lijn $vh \leq 0$.
 $A = vh$
 Zie invoer deel 12.
39. Een nummer van een lijn die m.b.v. de verhoudingsgetallen vh_{ij} verdeeld worden, valt buiten de range: $0 \leq \text{lijnummer} \leq \text{aantal lijnen}$ in het betreffende net.
 $B =$ aantal lijnen
 $C =$ lijnummer
 Zie invoer deel 13.

40. Een opgegeven verhoudingsgetal $vh_{ij} \leq 0$
A= vh_{ij}
Zie invoer deel 13.
41. Het nummer van een samengestelde lijn valt buiten de range:
 $0 \leq \text{lijnnummer} \leq \text{aantal lijnen in het betreffende net.}$
C= lijnnummer
Zie invoer deel 14.
42. Het verschil tussen de aantallen intervallen van twee opeenvolgende lijnen > 1 .
B= aantal intervallen voor bepaalde lijn.
C= aantal intervallen voor voorgaande lijn.
Zie invoer deel 1.
43. Het nummer van het eerste subnet $esn < 1$ of het nummer van het laatste subnet $(esn+asn-1)$ is groter dan het totaal aantal subnetten (tasn).
A= tasn
B= esn
C= asn
Zie invoer deel 1.

5. OPMERKINGEN

5.1. Dit programma bevat een aantal procedures, die ook daar buiten van belang kunnen zijn, zoals:

TOPOLOGY	: element generatie.
MG 01	: element (driekhoekig)- en knooppuntgeneratie.
KOR	: correcties op bestaande elementverdeling.
TRANSEL	: genereren van tussenpunten op elementribben.
SORTLINES	: recursief werkende sorteerprocedure.
FIG 3	: economisch plotten van elementverdelingen.
VOLG OMTREK	: verzamelen van de contourpunten van een constructie

5.2. Het is niet meer mogelijk om de programma-identificatie bij de plotfiguren te schrijven sinds het programma geschikt gemaakt is voor de Burroughs B6700. Zodra het R.C. de daarvoor benodigde software kan leveren wordt dit gerealiseerd. De tekst van maximaal drie regels wordt in het programma bewaard in de vorm van strings S1,S2 en S3[60], terwijl art het aantal zinvol gevulde strings aangeeft.

5.3. Het blijkt dat bij de operaties die op de elementverdeling kunnen worden uitgevoerd, maar ook bij het plotten van een elementverdeling, de arrays LE,X en Y niet zo interessant zijn. Veel belangrijker is dan een array dat de lijnen van de verdeling in gesorteerde vorm bevat. Bij een nieuwe opzet verdient het zeker aanbeveling om dit array samen te stellen zodra de arrays LE,X en Y bekend zijn, en gedurende de executie van het programma te bewaren.