

HP 48G

Versie 2020-04-17





Het Programma

Inhoud

Voorwoord	8
Verantwoordelijkheid	8
Deel 2. HET PROGRAMMA	9
0. Variabelen:	9
Globale variabelen:	9
Local variables used in the listing:	11
Speciale symbolen gebruikt in de programma's:	11
1a. RSTFLD (ReSeT FieLD)	12
Beschrijving:	12
Listing:	12
1b. RSTTN (ReSeT Target Numbers)	13
Beschrijving:	13
Listing:	13
1c. CRTFLD (CReaTe FieLD)	14
Beschrijving:	14
Listing:	14
1d. CRTSPCFLD (CReaTe SPeCial FieLD)	15
Beschrijving:	15
Listing:	15
Diagramma's:	16
2. TNFH, TNFD & TNFV (Target Number Field Horizontal, Diagonal en Vertical)	18
Beschrijving:	18
Listing:	18
2b. TNFD (Target Number Field Diagonal)	19
Listing:	19
2c. TNFV (Target Number Field Vertical)	20
Listing:	20
2d. CHKFLD (CHeCK FieLD)	21
Beschrijving:	21
Listing:	21
Diagramma's:	22
3a. MNFRQTN & MXFRQTN (MiNiMum FReQuency Target Number)	24
Beschrijving:	24
Listing:	24
3b. MXFRQTN (MaXiMum FReQuency Target Number)	24
Listing:	24
Diagramma's:	25

4. GENTN (GENerate Target Number)	26
Beschrijving:	26
Listing:	26
Diagram:	26
5a. NXTTN & PRVTN (NeXT Target Number)	27
Beschrijving:	27
Listing:	27
5b. PRVTN (PReVious Target Number)	27
Listing:	27
Diagramma's:	28
6. SCREEN (SCREEN of field to play)	29
Beschrijving:	29
Listing:	29
Diagram:	30
7a. TNSH, TNSD & TNSV (Target Number Solution Horizontal, Diagonal en Vertical)	31
Beschrijving:	31
Listing:	31
7b. TNSD (Target Number Solution Diagonal)	32
Listing:	32
7c. TNSV (Target Number Solution Vertical)	33
Listing:	33
7d. CHKSOLFLD (CHecK SOLution FieLD)	34
Beschrijving:	34
Listing:	34
Diagramma's:	35
8a. NXTSOL & PRVSOL (NeXT SOLution)	37
Beschrijving:	37
Listing:	37
8b. PRVSOL (PReVious SOLution)	37
Listing:	37
Diagramma's:	38
9. SOLSCREEN (SOLution SCREEN)	39
Beschrijving:	39
Listing:	39
Diagram:	41
10a. QFIELD & QFLD (Quality FIELD & Quality FieLD)	42
Beschrijving:	42
Listing:	42

10b. QFLD (Quality Field)	43
Listings:	43
Diagramma's:	44
3. MENU	47
0. TRIO	47
Beschrijving:	47
Listing:	47
1. Primair niveau Menu	48
Beschrijving:	48
Listing:	48
2. Secundair niveau Menu	49
Beschrijving:	49
Listings:	49
3. Tercair niveau Menu	52
Beschrijving:	52
Listings:	52
4. Quartair niveau Menu	53
Beschrijving:	53
Listings:	53

Voorwoord

In het eerste deel worden de spelregels en hoe TRIO te spelen op de HP 48 G/GX uitgelegd. Dit tweede deel is de gedetailleerde beschrijving en de code van het programma. In deel drie zijn hulpprogramma's weergegeven om het programma zo snel mogelijk te laten verlopen. Tot slot; in deel 1 (Appendix) zijn er printbare pagina's om TRIO zelf te spelen (zoals het in 1975 bedoeld was).

Verantwoordelijkheid

Hoewel dit werk met de grootste zorg en controle is uitgevoerd kan de schrijver/programmeur niet worden verantwoordelijk geacht voor eventuele fouten/schade, noch in de software, noch in de hardware van Uw HP 48 G/GX toestel.

Deel 2. HET PROGRAMMA

0. Variabelen:

Globale variabelen:

Een overzicht van de globale variabelen die gebruikt worden over de programmadelen heen. Deze globale variabelen kunnen bestaan uit lijsten of uit cijfers/getallen. Alle waarden hieronder vermeld zijn niet fictief, maar dienen wel als voorbeeld. Uiteraard zullen deze waarden/volgorde van de waarden, bij jou anders zijn.

'FIELD'

```
{ 1 5 7 2 6 4 5 8 3
  9 8 1 6 6 9 7 7 9 9
  8 1 8 4 2 6 5 5 4 3
```

```
5 8 5 6 2 3 4 7 1 2
2 6 7 3 4 1 4 3 3 2
}
```

FIELD = 137 bytes, een lijst met 49 cijferkaartjes in willekeurige volgorde.

'FRQTN'

```
{ 9 4 7 7 9 10 14
  11 19 12 12 10 15 9
  15 8 16 3 16 9 6 8
  8 2 8 4 7 7 5 8 6
```

```
7 8 8 5 5 3 9 3 4 4
5 4 4 6 3 4 2 2
}
```

FRQTN = ± 252 bytes, een lijst met de frequentie van de 50 doelgetallen, in volgorde van voorkomen in **TNLST**.

'min'

```
2
```

min = 18 bytes, minimum frequentie, uit **FRQTN**.

'max'

```
19
```

max = 18 bytes, maximum frequentie, uit **FRQTN**.

'TNLST'

```
{ 1 10 4 37 38 50 2
 7 16 8 17 11 14 41
33 13 5 35 28 6 45
42 24 15 3 23 48 27
```

```
40 32 22 46 30 47
36 19 18 31 21 43 9
25 49 39 29 26 12
34 44 20 }
```

TNLST = 468 bytes, lijst met de 50 doelgetallen (**TN**) in willekeurige volgorde.

'pntr'

2

pntr = 19 bytes, pointer, aanduiding welk **TN** uit de lijst **TNLST**.

'tn'

10

tn = 17 bytes, doelgetal.

'frq'

12

frq = 18 bytes, frequentie van doelgetal **tn**.

In dit voorbeeld zie je de **pntr** = 2; in de lijst **TNLST** zie je als tweede getal 10 (**tn**) en uit de lijst **FRQTN** zie je op de 10^{de} plaats 12 (= **frq**).

'spntr'






1

spntr = 20 bytes, pointer, gebruikt bij het geven van de oplossingen (combinaties van een doelgetal).

Local variables used in the listing:

- b: begin, eerste cijfer van een combinatie
- e: einde, laatste cijfer van een combinatie
- hs: horizontale som (van een combinatie)
- ds: diagonale som (van een combinatie)
- vs: verticale som (van een combinatie)
- k: teller aantal trios van een doelgetal
- q: zelf gekozen minimum aantal combinaties

Speciale symbolen gebruikt in de programma's:

- : staat voor een spatie in tekst-delen (tussen "")
- : staat voor een line feed in een tekst-deel ("") die op de volgende regel verdergaat.
- : stands for code 127 in the **CHARS**-editor

1a. RSTFLD (ReSeT FieLD)

Beschrijving:

Maakt de lijst **FIELD** aan, met het juiste aantal cijferkaartjes (zoals het originele spel het voorschrijft). Dit programma moet slechts uitgevoerd als de lijst **FIELD** beschadigd of verloren gegaan is.

Listing:

'RSTFLD'

```
« CLEAR { 1 1 1 1 1
2 2 2 2 2 3 3 3 3
3 3 4 4 4 4 4 5 5
5 5 5 5 6 6 6 6 6
```

```
7 7 7 7 7 8 8 8 8 8
9 9 9 9 } 'FIELD'
STO
»
```

RSTFLD = 167 bytes

12 April 2020

Zie diagram 1a.

1b. RSTTN

(ReSeT Target Numbers)

Beschrijving:

Maakt de lijst **TNLST** aan, met het juiste aantal TN's (*doelgetallen*), zoals het originele spel het voorschrijft. Dit programma moet slechts uitgevoerd worden als de lijst **TNLST** beschadigd of verloren gegaan is.

Listing:

'RSTTN'

```
« CLEAR 1 50
  FOR i i
    NEXT DUP →LIST
  'TNLST' STO
»
```

RSTTN = 70 bytes

02 April 2020

Zie diagram 1b.

1c. CRTFLD (CReaTe FieLD)

Beschrijving:

Dit deelprogramma – opgeroepen door **CRTSPCFLD** – genereert een **FIELD** (*speelveld*) met de 49 cijferkaartjes in willekeurige volgorde.

Listing:

'CRTFLD'

```
« CLEAR 0 RDZ FIELD
OBJ→ 1 SWAP
  FOR i 49 i - RAND
* IP i + ROLL 3
  STEP 49 →LIST
'FIELD' STO
»
```

CRTFLD = 120 bytes

23 March 2020

Zie diagram 1c.

Opmerking:

Vervang 3 **STEP** (waarbij er 17 *cijferkaartjes* verwisseld worden) door:

- 1 **STEP:** om alle 49 *cijferkaartjes* te wisselen (efficiënt, maar langzaam)
- 2 **STEP:** 25 wissels
- 3 **STEP:** 17 wissels
- 4 **STEP:** 13 wissels
- 6 **STEP:** 9 wissels
- 8 **STEP:** 7 wissels
- 12 **STEP:** 5 wissels
- 16 **STEP:** 4 wissels
- 24 **STEP:** 3 wissels
- 49 **STEP:** 2 wissels (sneller, maar minder efficiënt)

1d. CRTSPCFLD (CReaTe SPeCial FieLD)

Beschrijving:

Dit programmadeel zoekt tot er een **FIELD** (*speelveld*) met de 49 willekeurige *cijferkaartjes* gegenereerd is die aan specifieke eigenschappen voldoet. Zo kan er bijvoorbeeld een **FIELD** gegenereerd worden waarvan de minimum *frequentie* exact 2 is (**UNTIL min 2 ==**), de variabele **min** is reeds als globale variabele gedeclareerd.

Listing:

'CRTSPCFLD'

```
« 1 TICKS → q take
tm
  « CLEAR CLLCD
  " _Please_Wait_..."
  1 DISP
  " _Looking_for_Min._"
  q + 2 DISP
  DO CRTFLD
  CHKFLD MNFRQTN
  " _Min._" min +
  "←
  " _Take_" + take +
  "←
  " _Time_" + TICKS
```

```
tm - B→R 8192 /
take / 100000 * IP
100000 / + 4 DISP 3
FREEZE 1 'take'
STO+
  UNTIL min q ==
  Min 5 ≥ OR
  END MXFRQTN ""
1 DISP
" _ _ _ _ Found._ _ _ _"
2 DISP
  »
»
```

CRTSPCFLD = 393 bytes

17 April 2020

Zie diagram 1d.

Diagramma's:

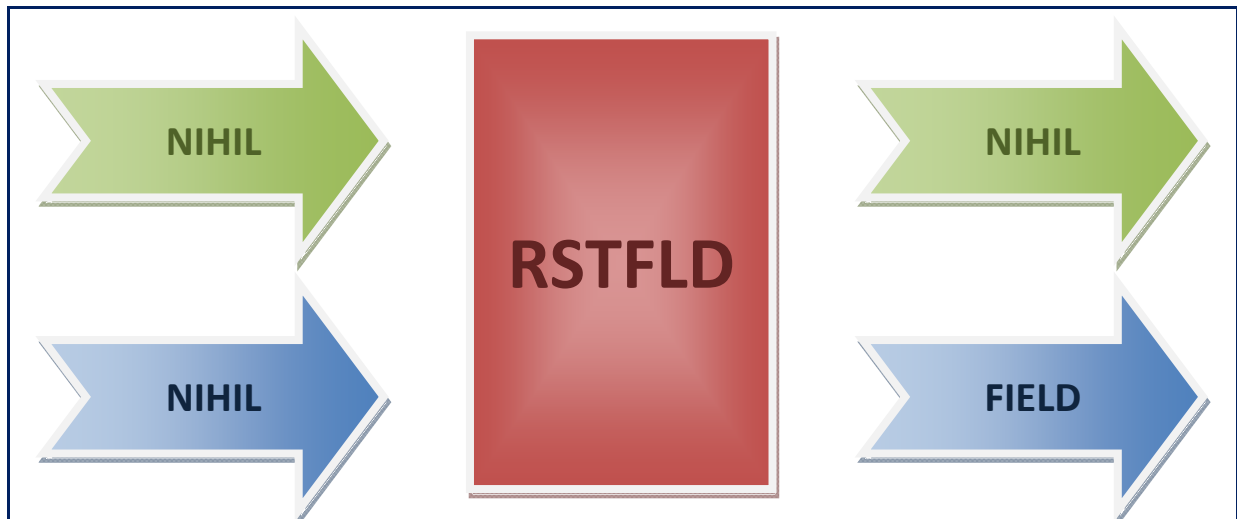


Diagram 1a: RSTFLD, reset field

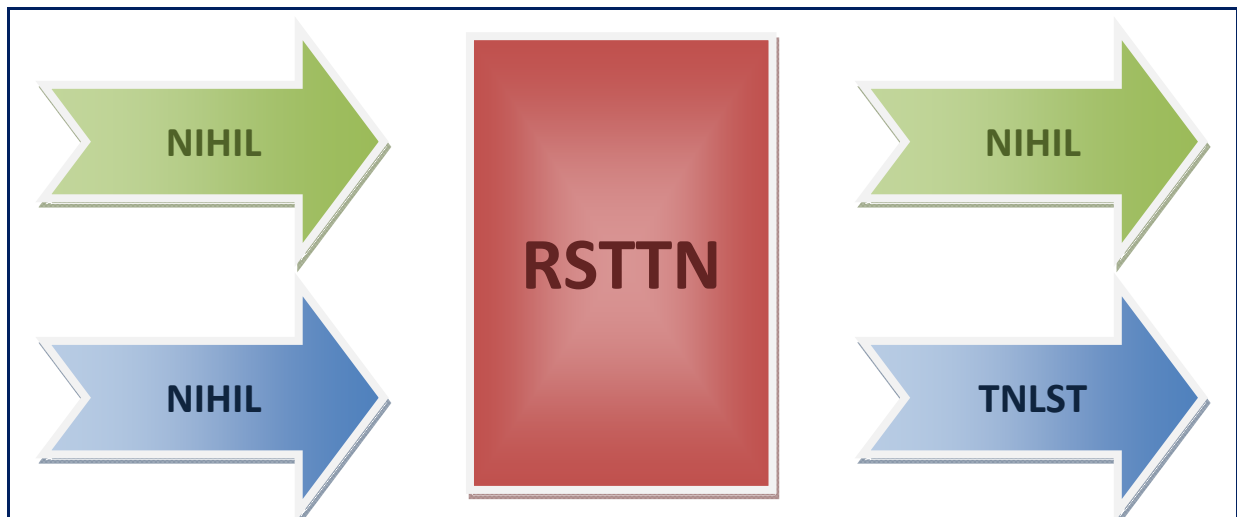


Diagram 1b: RSTTN, reset TN, reset de doelgetallen

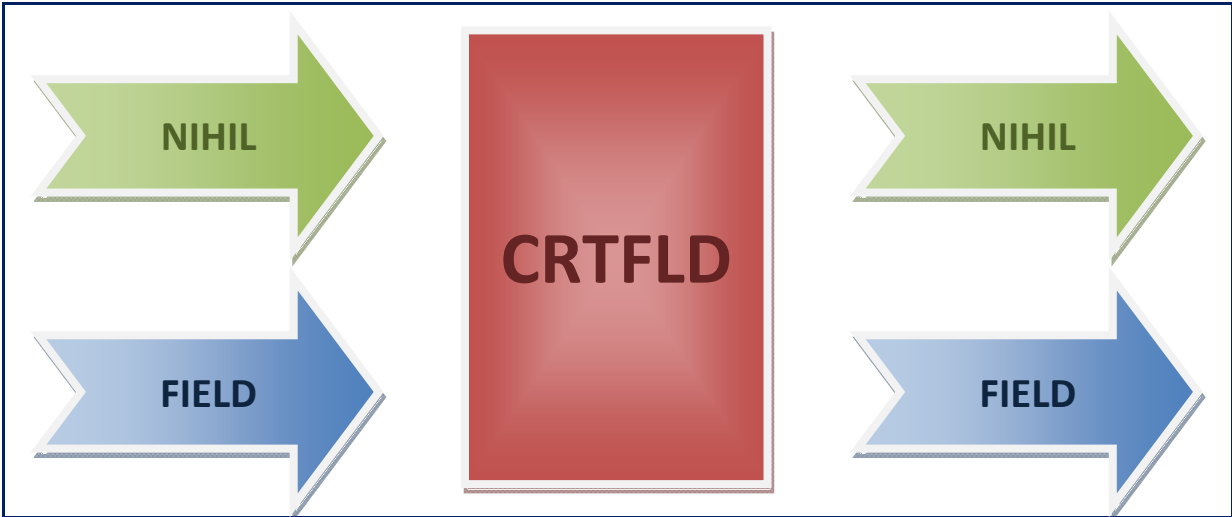


Diagram 1c: CRTFLD, create field

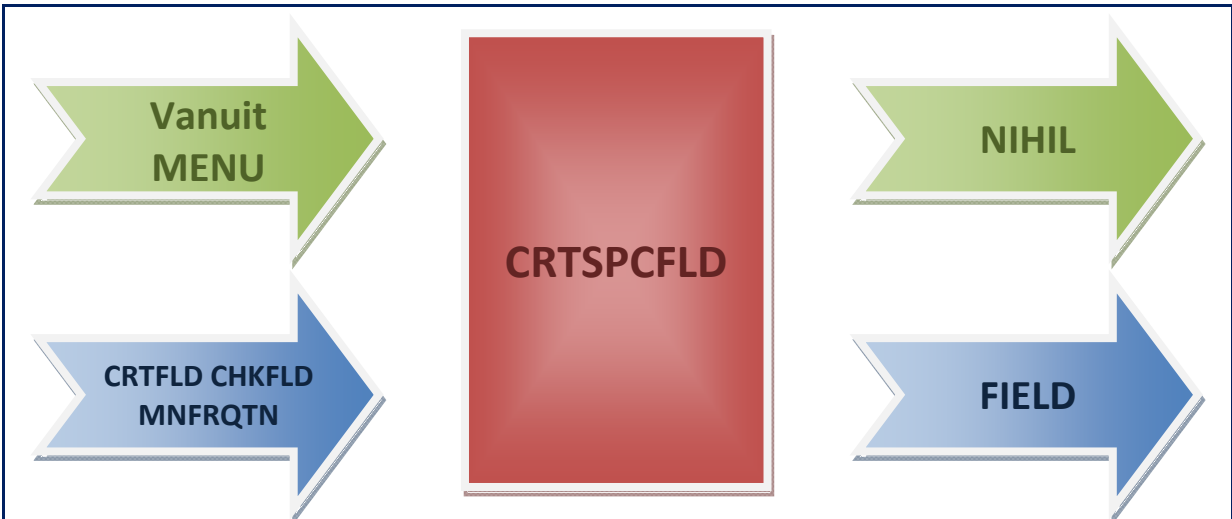


Diagram 1d: CRTSPCFLD, create special field

2. TNFH, TNFD & TNFV (Target Number Field Horizontal, Diagonal en Vertical)

Beschrijving:

Deze 3 programmadelen – allen opgeroepen vanuit **CHKFLD** – berekenen van alle mogelijke *combinaties* het resultaat. Het aantal malen dat een bepaalde *combinatie* eenzelfde resultaat heeft (*frequentie*) wordt in de lijst **FRQTN** opgeslagen. **TNFH** berekent de *horizontale combinaties* terwijl **TNFD** en **TNFV** resp. de *verticale* en *diagonale combinaties* berekenen.

Listing:

'TNFH'

```
« 1 5 1 → b e h s
  « CLEAR FIELD
REVLIST OBJ→ DROP 0
42
  FOR r b r + e r
+
    FOR s 1 SF s
PICK s 1 + h s +
PICK s 2 + h s 2 * +
PICK 3 DUPN 3 DUPN
ROT ROT 3 DUPN 1 4
  FOR i *
    IF 1 FS?C
    THEN +
    ELSE SWAP
-
  END DUP
DUP2
```

```
IF 0 >
SWAP 50 ≤ AND
THEN
FRQTN SWAP GET 1 +
FRQTN 3 ROLLD PUT
'FRQTN' STO
ELSE
DROP2
END
IF i 2 ==
THEN 1 SF
END
NEXT
NEXT 7
STEP
»
»
```

TNFH= 372 bytes

23 March 2020

Zie diagram 2a.

2b. TNFD

(Target Number Field Diagonal)

Listing:

'TNFD'

```
« → b e d s
  « CLEAR FIELD
REVLIST OBJ→ DROP 0
28
  FOR r b r + e r
+
    FOR s 1 S F s
PICK s 1 + d s +
PICK s 2 + d s 2 * +
PICK 3 DUPN 3 DUPN
ROT ROT 3 DUPN 1 4
  FOR i *
    IF 1 FS?C
    THEN +
    ELSE SWAP
-
    END DUP
DUP2
```

```
IF 0 >
SWAP 50 ≤ AND
  THEN
FRQTN SWAP GET 1 +
FRQTN 3 ROLLD PUT
'FRQTN' STO
  ELSE
DROP2
  END
  IF i 2 ==
  THEN 1 SF
  END
  NEXT
  NEXT 7
STEP
»
»
```

TNFD = 364 bytes

23 March 2020

Zie diagram 2b.

2c. TNFV

(Target Number Field Vertical)

Listing:

'TNFV'

```
« 1 35 7 → b e vs
  « CLEAR FIELD
  REVLIST OBJ→ DROP b
  e
    FOR s 1 SF s
    PICK s 1 + vs +
    PICK s 2 + vs 2 * +
    PICK 3 DUPN 3 DUPN
    ROT ROT 3 DUPN 1 4
      FOR i *
        IF 1 FS?C
          THEN +
          ELSE SWAP -
          END DUP
      DUP2
```

```
      IF 0 > SWAP
50 ≤ AND
      THEN FRQTN
      SWAP GET 1 + FRQTN
      3 ROLLD PUT 'FRQTN'
      STO
      ELSE DROP2
      END
      IF i 2 ==
      THEN 1 SF
      END
      NEXT
    NEXT
  »
»
```

TNFV = 341 bytes

February 2020

Zie diagram 2c.

2d. CHKFLD

(CHecK FieLD)

Beschrijving:

Dit programmadeel, roept de programmadelen **TNFH**, **TNFD** (2x) en **TNFV** op.

Listing:

'CHKFLD'

```
« CLEAR { } 1 50
  START 0 +
  NEXT 'FRQTN' STO
TNFV TNFH 3 7 6
TNFD 1 5 8 TNFD
CLEAR
»
```

CHKFLD = 112 bytes

08 April 2020

Diagramma's:

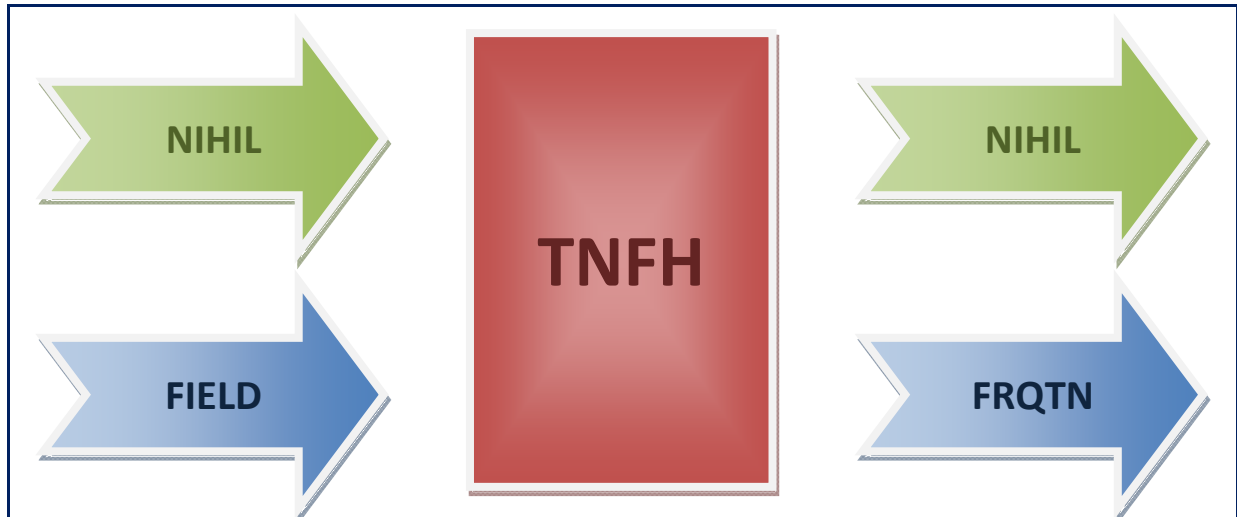


Diagram 2a: TNFH, horizontaal

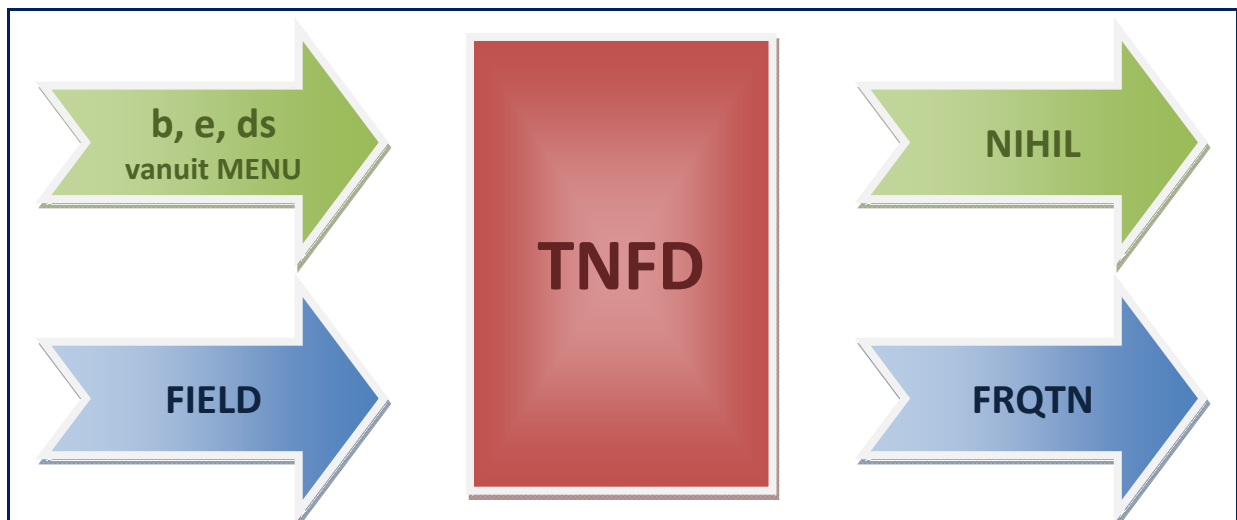


Diagram 2b: TNFD, diagonaal

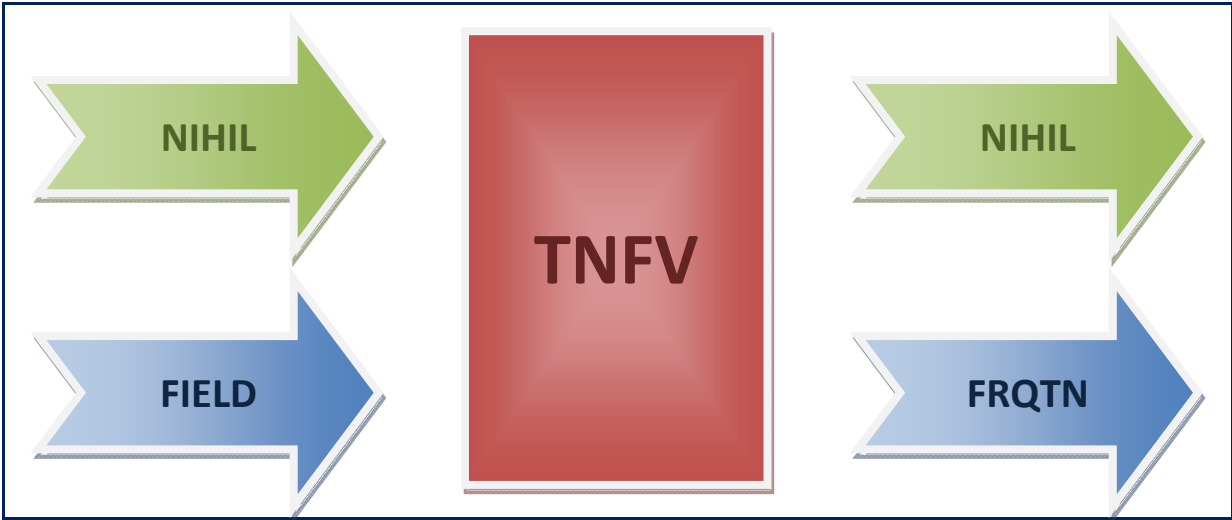


Diagram 2c: TNFV, verticaal

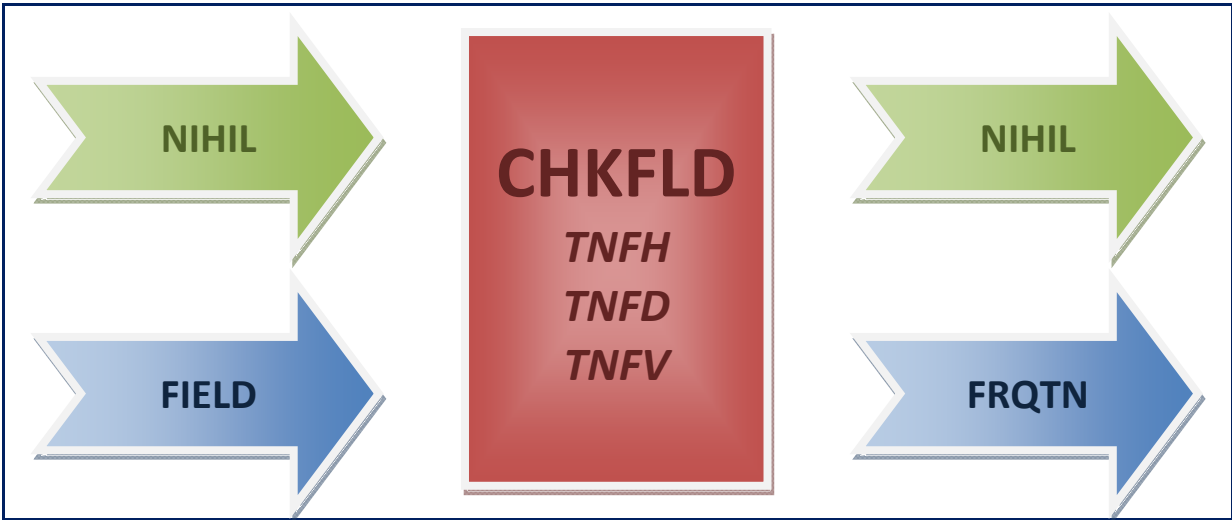


Diagram 2d: Check Field

3a. MNFRQTN & MXFRQTN (MiNimum FReQuency Target Number)

Beschrijving:

Dit programmadeel bepaalt de minimum *frequentie* van alle Target Numbers. Het resultaat wordt in de globale variabele **min** opgeslagen. Programmadeel 3b (**MXFRQ**) voert hetzelfde uit maar om de maximumwaarde te bepalen, deze waarde wordt in **max** opgeslagen.

Listing:

'MNFRQTN'

```
« FRQTN
  « MIN
  » STREAM 'min'
STO
»
```

MNFRQTN = 65 bytes

26 March 2020

3b. MXFRQTN (MaXimum FReQuency Target Number)

Listing:

'MXFRQTN'

```
« FRQTN
  « MAX
  » STREAM 'max'
STO
»
```

MXFRQTN = 65 bytes

26 March 2020

Diagramma's:

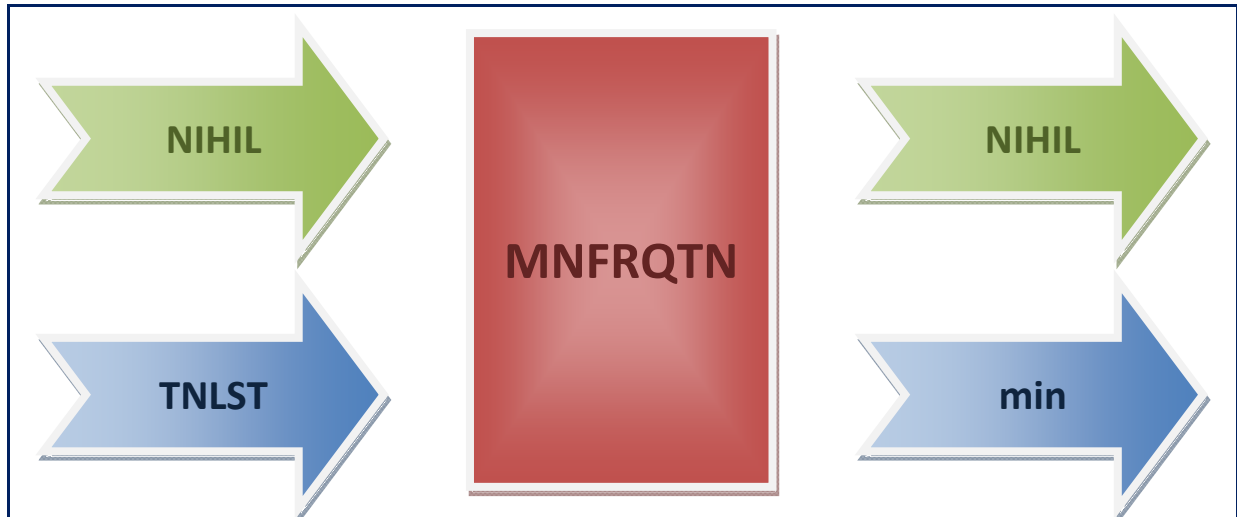


Diagram 4a: MNFRQ, minimum frequentie

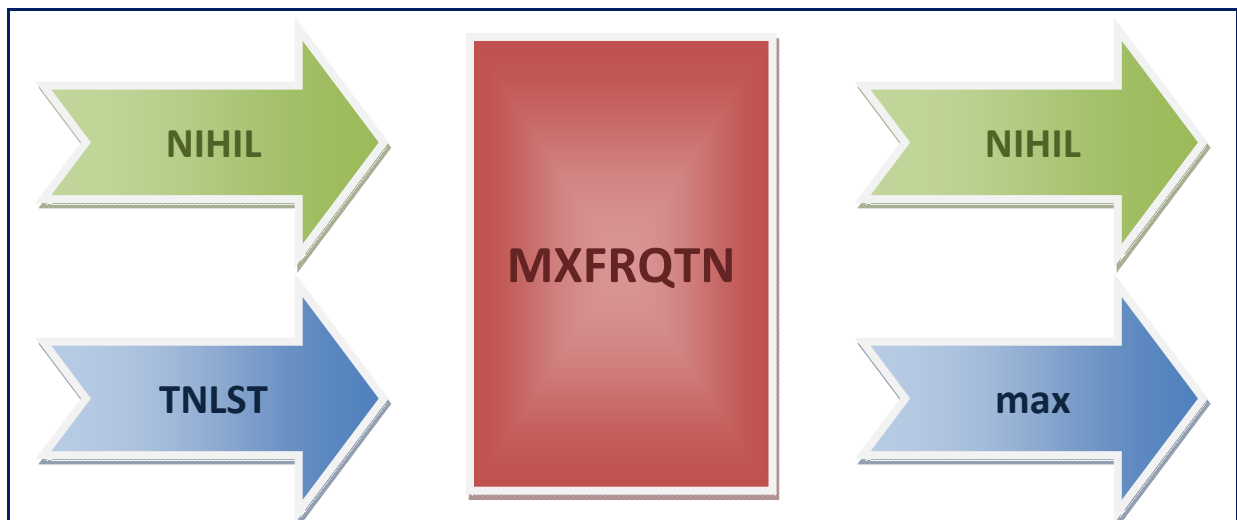


Diagram 4b: MXFRQ, maximum frequentie

4. GENTN

(GENerate Target Number)

Beschrijving:

Genereert een lijst met de 50 Target Numbers (*doelgetallen*) in een willekeurige volgorde.

Listing:

'GENTN'

```
« CLEAR 0 RDZ TNLST
  OBJ→ 1 SWAP
  FOR i 50 i - RAND
  * IP i + ROLL 7
  STEP 50 →LIST
  'TNLST' STO 1 DUP
```

```
'pntr' STO TNLST
SWAP GENT DUP 'tn'
STO FRQTN SWAP GET
'frq' STO
»
```

GENTN = 195 bytes

2! May 2020

Zie diagram 3.

Opmerking: vervang **7 STEP** (8 doelgetallen wisselen; sneller) door **NEXT** (alle doelgetallen wisselen; langzamer) Andere mogelijkheden (zoals bij **CTRFLD**) zijn er hier niet.

Diagram:

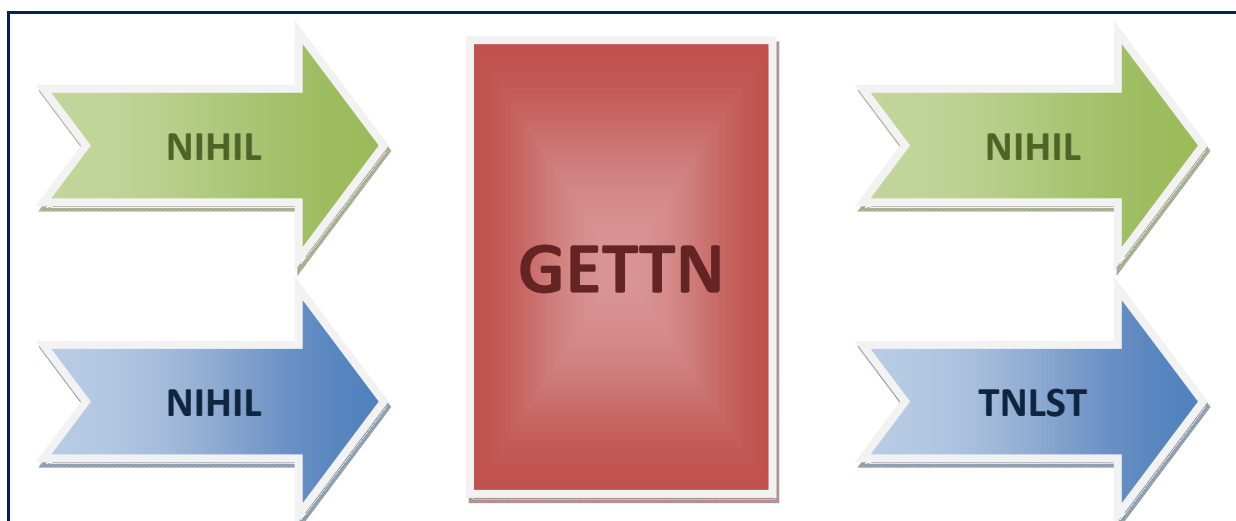


Diagram 3: GETTN, genereren van lijst met doelgetallen in willekeurige volgorde

5a. NXTTN & PRVTN

(NeXT Target Number)

Beschrijving:

Dit programmadeel (**NXTTN**) toont het volgende Target Number (*doelgetal*) uit de lijst **TNLST** met *doelgetallen* in willekeurige volgorde. **PRVTN** toont het vorige *doelgetal*, uit diezelfde lijst.

Listing:

'NXTTN'

```
« TNLST pntr 1 +
DUP 51
  IF ≥
  THEN DROP 1
  "___GAME__OVER↵
  _____→_50/50"
```

```
MSGBOX
  END DUP 'pntr'
  STO GET DUP 'tn'
  STO FRQTN SWAP GET
  'frq' STO
»
```

NXTTN = 171 bytes

09 April 2020

5b. PRVTN

(PReVIOUS Target Number)

Listing:

'PRVTN'

```
« TNLST pntr 1 -
DUP 0
  IF ≤
  THEN DROP 1
  "___BEGIN__GAME↵
  _____→_1/50"
```

```
MSGBOX
  END DUP 'pntr'
  STO GET DUP 'tn'
  STO FRQTN SWAP GET
  'frq' STO
»
```

PRVTN = 164 bytes

09 April 2020

Diagramma's:

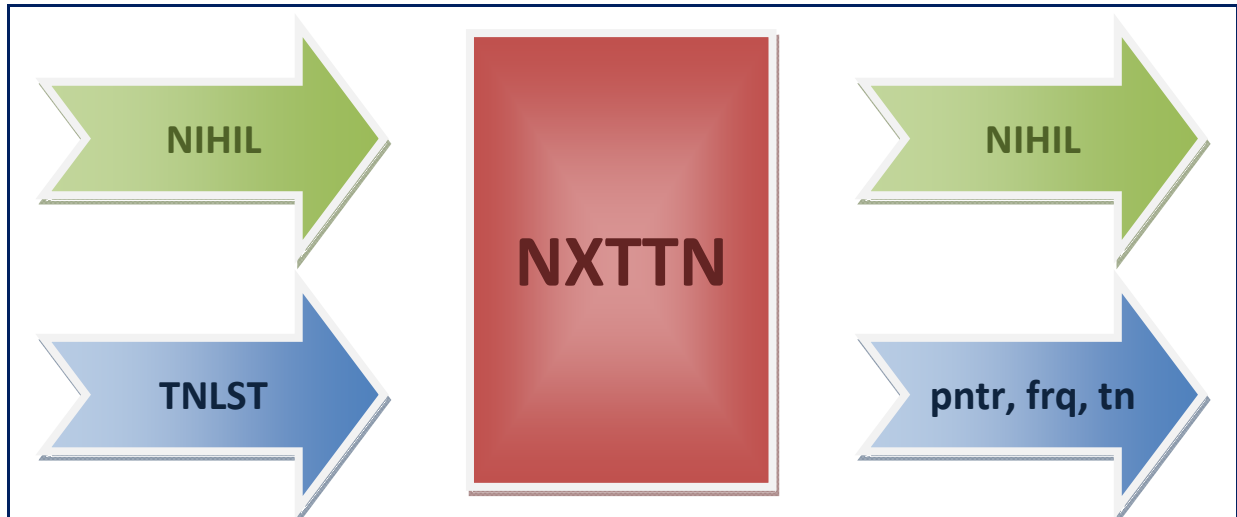


Diagram 5a: Next TN, volgend doelgetal

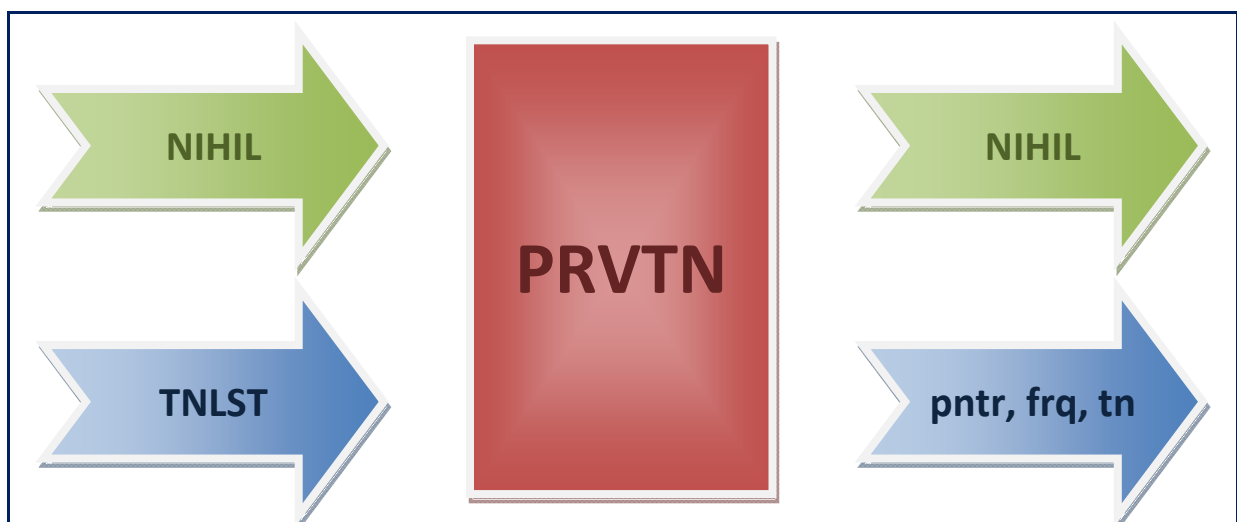


Diagram 5b: Previous TN, vorig doelgetal

6. SCREEN

(SCREEN of field to play)

Beschrijving:

Hier wordt het *speelveld* **FIELD** afgedrukt op het scherm. Het rechterdeel van het scherm bevat het *speelveld*; het linkerdeel, het *statusdeel*, bevat volgende gegevens:

- Het te zoeken Target Number (uit **tn** gehaald);
- Het zoveelste Target Number uit de lijst **TNLST** met *doelgetallen* (uit **pntr** gehaald);
- Het puntenaantal dat de speler krijgt die de oplossing (*combinatie*) vindt (berekend);
- De minimum *frequentie* van voorkomen in het *speelveld* (uit **min** gehaald);
- De *frequentie* van voorkomen in het *speelveld* (uit **frq** gehaald);
- De maximum *frequentie* van voorkomen in het *speelveld* (uit **max** gehaald);

Listing:

'SCREEN'

```
« CLLCD FIELD
REVLIST OBJ→ DROP
  IF min 0 ==
  THEN 0 'frq' STO
  END 0 6
  FOR i
    CASE i 0 ==
      THEN "TN:_"
      IF tn 10 <
      THEN "_" +
      END tn +
    " | " +
      END i 1 ==
      THEN
        IF max 1 +
frq - 10 <
        THEN "_"
        ELSE ""
        END max 1 +
frq - + "_pt" +
```

```
IF max 1 +
frq - 1 ≠
  THEN "s | "
  ELSE "_" | "
  END +
  END i 2 ==
  THEN "→"
  IF pntr 10
<
  THEN "_" +
  END pntr +
"/50 | " +
  END i 4 ==
  THEN "Mn._"
  IF min 10 <
  THEN "_" +
  END min +
  " | " +
  END i 5 ==
  THEN "#_ _ _"
```

```

        IF frq 10 <
        THEN " " +
        END frq +
"|" +
        END i 6 ==
        THEN "Mx. "
        IF max 10 <
        THEN " " +
        END max +

```

```

"|" +
        END " " +
        END 1 7
        FOR j " " +
        SWAP +
        NEXT i 1 + DISP
        NEXT 3 FREEZE
»

```

SCREEN = 756 bytes

10 April 2020

Remarks:

" " number of spaces between the " ", so " " means one space; " " means two spaces. And so on.

"|", is $\alpha \rightarrow J$, alpha / right shift / J. The colour red in the listing is to make is more visible.

Diagram:

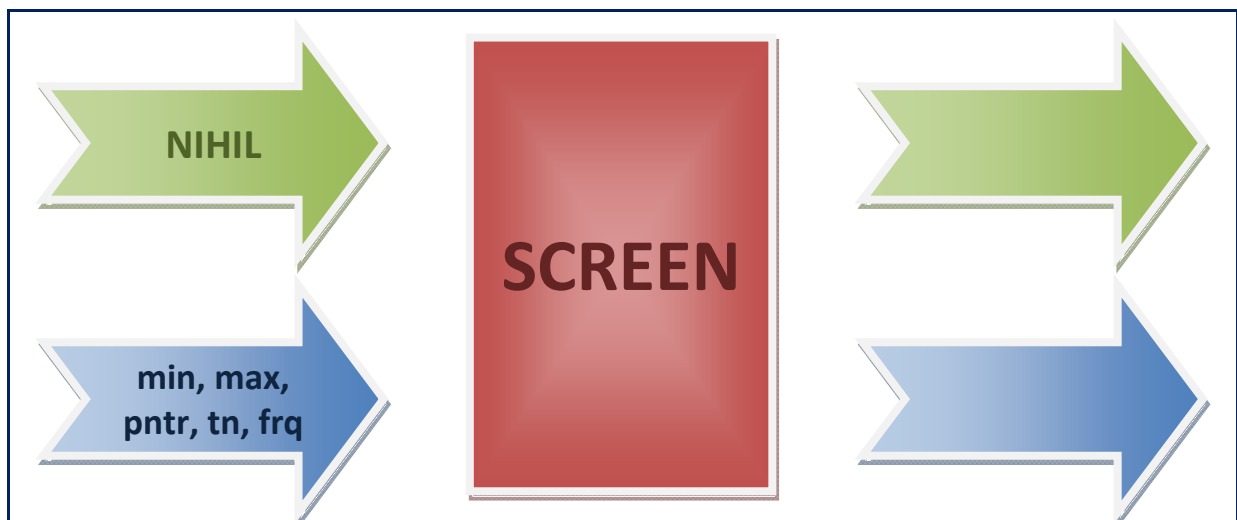


Diagram 6: SCREEN, uitvoer naar het scherm.

7a. TNSH, TNSD & TNSV

(Target Number Solution Horizontal, Diagonal en Vertical)

Beschrijving:

Deze 3 programmadelen – allen opgeroepen vanuit **CHKSOLFLD** – berekenen van alle mogelijke *combinaties* de posities van de oplossingen en slaat deze op in de lijst **TNSOLLST**. Deze programmadelen worden opgeroepen vanuit **CHKSOLFLD** (Zie 7d.). **TNSH** berekent de *horizontale combinaties* terwijl **TNSD** en **TNSV** resp. de *verticale* en *diagonale combinaties* berekenen.

Listing:

'TNSH'

```
« 3 7 -1 → b e h s
« 0 42
  FOR r b r + e r
+
  FOR s 1 SF
FIELD s GET FIELD s
hs + GET FIELD s hs
2 * + GET 3 DUPN 3
DUPN ROT ROT 3 DUPN
1 4
  FOR i *
    IF 1 FS?C
    THEN +
    ELSE SWAP
-

```

```
END
IF tn ==
THEN
SOLLST s hs 2 * + +
s hs + + s +
'SOLLST' STO
END
IF i 2 ==
THEN 1 SF
END
NEXT
NEXT 7
STEP
»
»

```

TNSH = 359 bytes

23 March 2020

7b. TNSD

(Target Number Solution Diagonal)

Listing:

'TNSD'

```
« → b e d s
« 0 28
  FOR r b r + e r
+
  FOR s 1 SF
FIELD s GET FIELD s
ds + GET FIELD s ds
2 * + GET 3 DUPN 3
DUPN ROT ROT 3 DUPN
1 4
  FOR i *
    IF 1 FS?C
    THEN +
    ELSE SWAP
-

```

```
END
  IF tn ==
  THEN
SOLLST s ds 2 * + +
s ds + + s +
'SOLLST' STO
  END
  IF i 2 ==
  THEN 1 SF
  END
  NEXT
  NEXT 7
STEP
»
»

```

TNSD = 351 bytes

23 March 2020

7c. TNSV

(Target Number Solution Vertical)

Listing:

'TNSV'

```
« 1 35 7 → b e vs
  « b e
    FOR s 1 SF
FIELD s GET FIELD s
vs + GET FIELD s vs
2 * + GET 3 DUPN 3
DUPN ROT ROT 3 DUPN
1 4
  FOR i *
    IF 1 FS?C
      THEN +
      ELSE SWAP -
    END
```

```
IF tn ==
  THEN SOLLST
s vs 2 * + + s vs +
+ s + 'SOLLST' STO
  END
  IF i 2 ==
    THEN 1 SF
  END
  NEXT
NEXT
»
»
```

TNSV = 328 bytes

23 March 2020

7d. CHKSOLFLD

(CHecK SOLution FieLD)

Beschrijving:

Dit programmadeel, roept de programmadelen **TNSH**, **TNSD** (2x) en **TNSV** op.

Listing:

'CHKSOLFLD'

```
« { } 'SOLLST' STO
TNSV TNSH 3 7 6
TNSD 1 5 8 TNSD 1
'spntr' STO SOLLST
1 GETI ROT ROT GET
DUP2 2 * SWAP -
»
```

SCKSOLFLD = 144 bytes

January 2020

Info:

TNSV: vertical check (↓ and ↑)

TNSH: horizontal check (→ and ←)

3 6 7 TNSD:diagonal check (↙ and ↗)

1 5 8 TNSD:diagonal check (↘ and ↖)

Diagramma's:

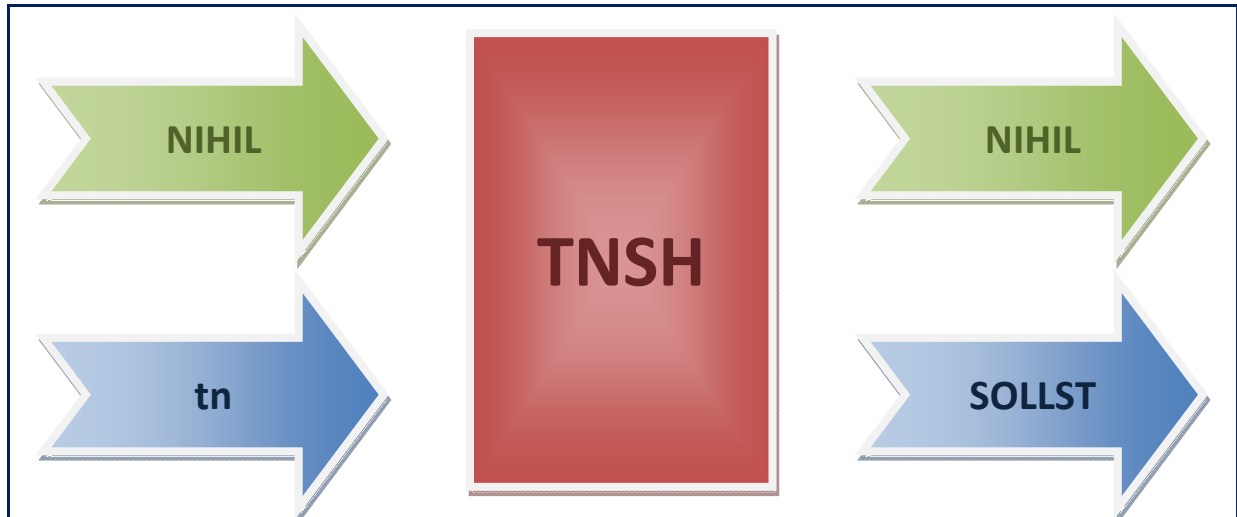


Diagram 7a: TNSH, horizontaal

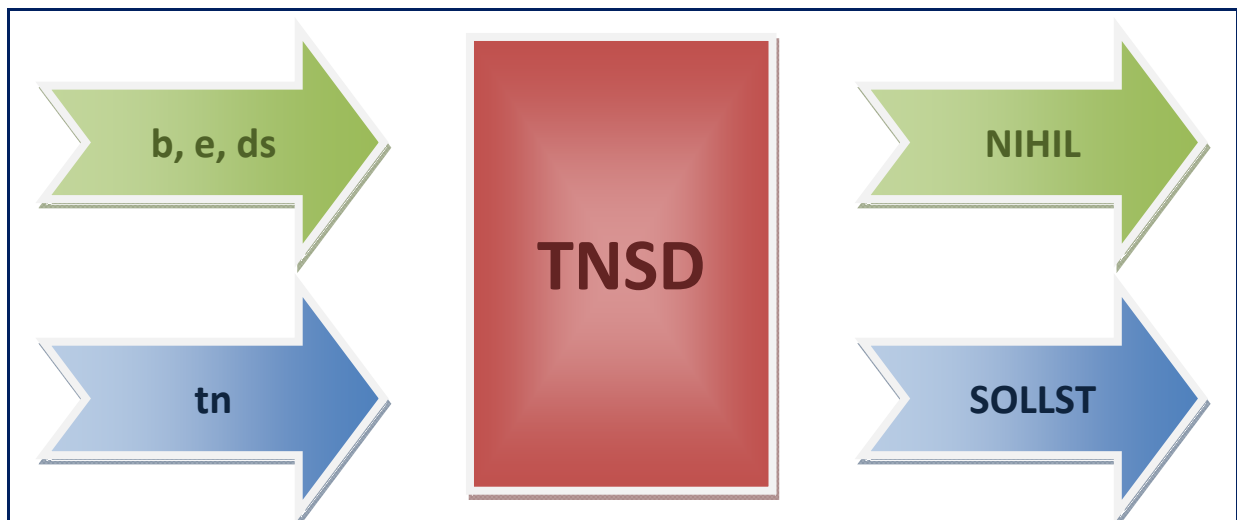


Diagram 7b: TNSD, diagonaal

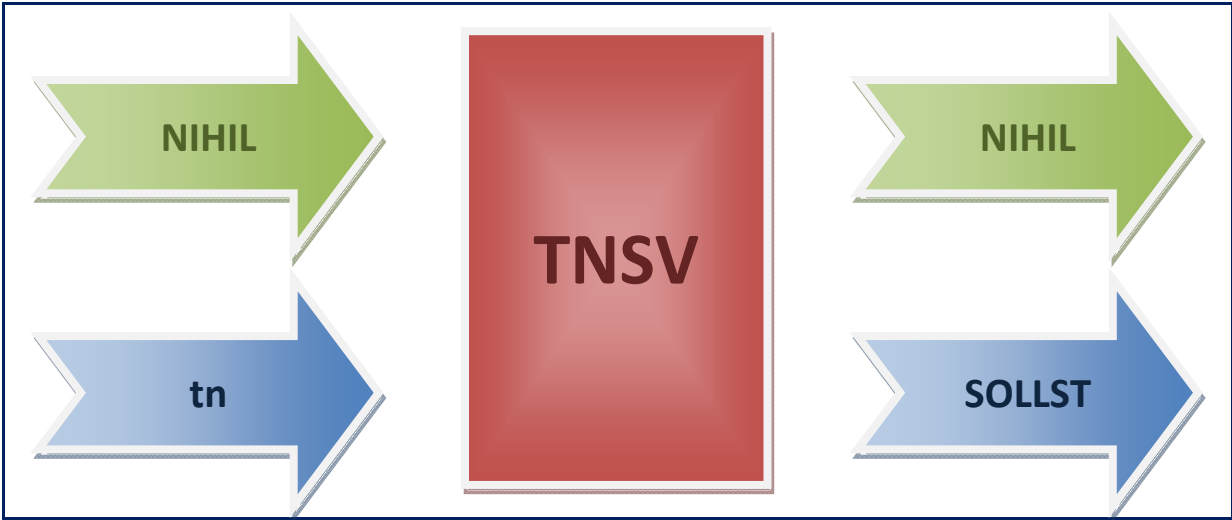


Diagram 7c: TNSV, vertikaal

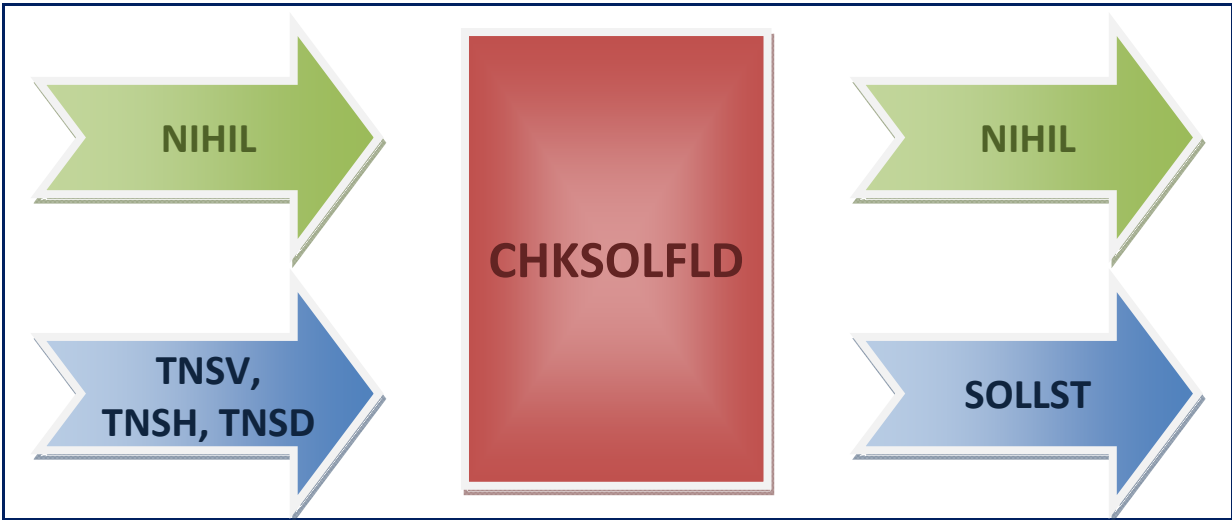


Diagram 8: CHKSOLFLD

8a. NXTSOL & PRVSOL (NeXT SOLution)

Beschrijving:

Dit programmadeel (**NXTSOL**) toont de volgende combinatie met oplossingen uit de lijst **SOLLST**. **PRVSOL** toont de vorige combinatie, uit diezelfde lijst.

Listing:

'NXTSOL'

```
« CLEAR SOLLST SIZE
3 / → k
  « spntr 1 +
    IF DUP k 1 + ==
      THEN k -
    END 'spntr' STO
```

```
SOLLST spntr 1 - 3
* 1 + GETI ROT ROT
GET DUP2 2 * SWAP -
  »
»
```

NXTSOL = 171 bytes

09 April 2020

8b. PRVSOL (PReVious SOLution)

Listing:

'PRVSOL'

```
« CLEAR SOLLST SIZE
3 / → k
  « spntr 1 -
    IF DUP 0 ==
      THEN k +
    END 'spntr' STO
```

```
SOLLST spntr 1 - 3
* 1 + GETI ROT ROT
GET DUP2 2 * SWAP -
  »
»
```

PRVSOL = 164 bytes

09 April 2020

Diagramma's:



Diagram 8a: Next SOL, volgende combinatie

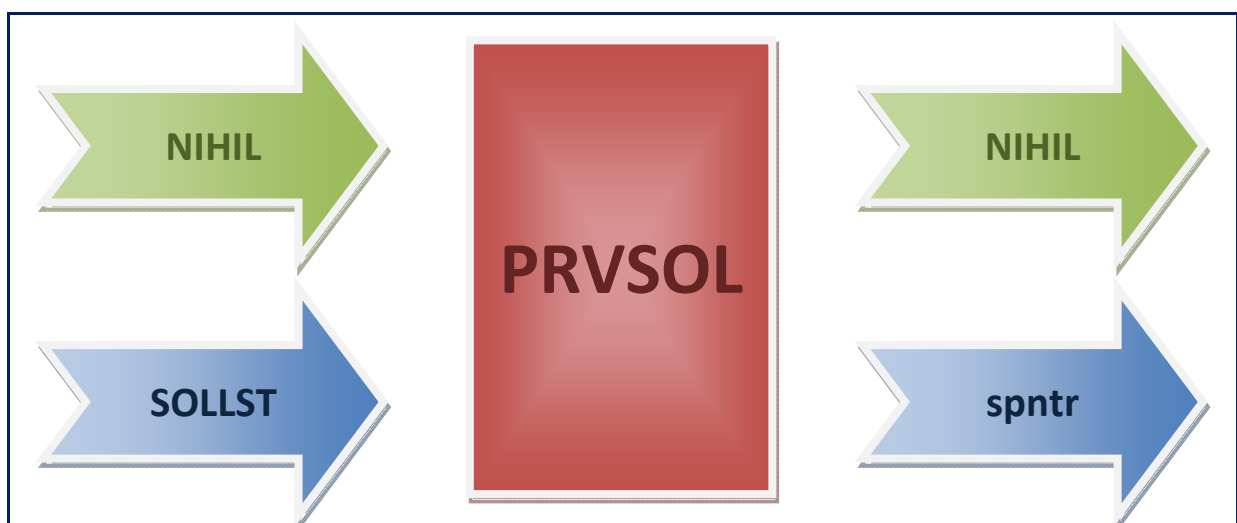


Diagram 8b: Previous SOL, vorig combinatie

9. SOLSCREEN (SOLution SCREEN)

Beschrijving:

Hier worden alle oplossingen voor een gegeven *doelgetal* op het scherm weergegeven. De gegevens worden uit de lijst **SOLLST** gehaald.

Listing:

'SOLSCREEN'

```
« SOLLST SIZE 3 / →
dg1 dg2 dg3 k
  « CLLCD FIELD
REVLIST OBJ→ DROP 0
6
  FOR i
    CASE i 0 ==
      THEN "TN:┐"
        IF tn 10
<
          THEN "┐"
+
          END tn +
"┐" +
          END i 1 ==
          THEN
            IF max 1
+ frq - 10 <
              THEN "┐"
              ELSE ""
              END max 1
+ frq - + "┐pt" +
            IF max 1
+ frq - 1 ≠
```

```
      THEN "s┐"
      ELSE "┐┐"
      END +
    END i 2 ==
    THEN "→"
      IF spntr
10 <
        THEN "┐"
+
        END spntr
+ "/" +
        IF k 10 <
          THEN "┐"
+
          END k +
"┐" +
        END i 4 ==
        THEN "Mn.┐"
          IF min 10
<
            THEN "┐"
+
            END min +
"┐" +
```

```

        END i 5 ==
        THEN "#_""
            IF frq 10
<
                THEN "_"
+
                    END frq +
"|" +
        END i 6 ==
        THEN "Mx._"
            IF max 10
<
                THEN "_"
+
                    END max +
"|" +
        END
    
```

SOLSCREEN = 913 bytes

```

"_____|"
        END 1 7
        FOR j
            CASE i 7 *
j + DUP dg1 == SWAP
DUP dg2 == SWAP dg3
== OR OR
                THEN "_"
+ SWAP +
            END SWAP
DROP "_.•" +
        END
        NEXT i 1 +
DISP
        NEXT 3 FREEZE
>>
>>
    
```

10 April 2020

Diagram:

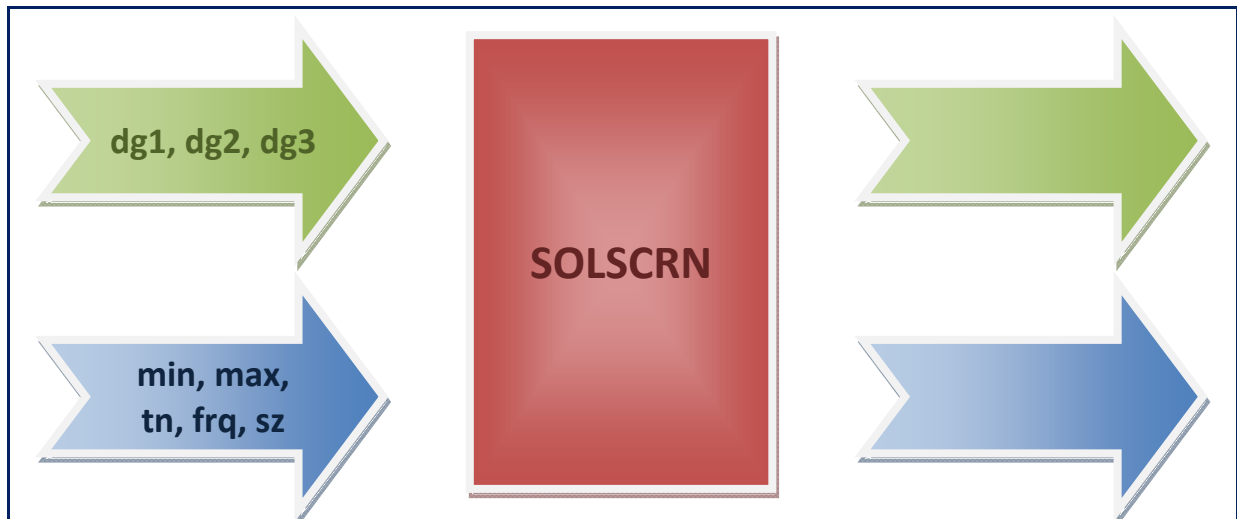


Diagram 9: SOLSCREEN: uitvoer naar het scherm

10a. QFIELD & QFLD

(Quality FIELD & Quality FieLD)

Beschrijving:

Beide (QFIELD & QFLD) programmadelen controleren of een zelf ingevoerd *speelveld* aan de voorwaarden voldoet. Zie hiervoor “*Inhoud van de doos (om als gezelschapsspel te spelen)*”, in deel 1.

Listing:

‘QFIELD’

```
« CLLCD FIELD SORT
REVLIST

  IF OBJ→ 49 ==
  THEN + + + + 5 ==
DEPTH ROLLD + + + +
+ 12 == DEPTH ROLLD
+ + + + + 18 ==
DEPTH ROLLD + + + +
+ 24 == DEPTH ROLLD
+ + + + + 30 ==
DEPTH ROLLD + + + +
+ 36 == DEPTH ROLLD
+ + + + + 35 == DEPTH
```

```
ROLLD + + + + 40 ==
DEPTH ROLLD + + +
36 == DEPTH ROLLD
AND AND AND AND AND
AND AND AND
  IF
  THEN 1
  ELSE 0
  END
ELSE CLEAR
END
»
```

QFIELD = 365 bytes

15 April 2020

Zie diagram 10a.

10b. QFLD (Quality FieLD)

Listings:

‘QFLD’

```
« CLLCD FIELD OBJ→  
1 SWAP 1 -  
  START +  
  NEXT 236  
  IF ==  
  THEN  
    IF QFIELD
```

```
      THEN 1  
      ELSE 0  
      END  
    ELSE 0  
    END  
  »
```

QFLD = 105 bytes

15 April 2020

Zie diagram 10b.

Diagramma's:

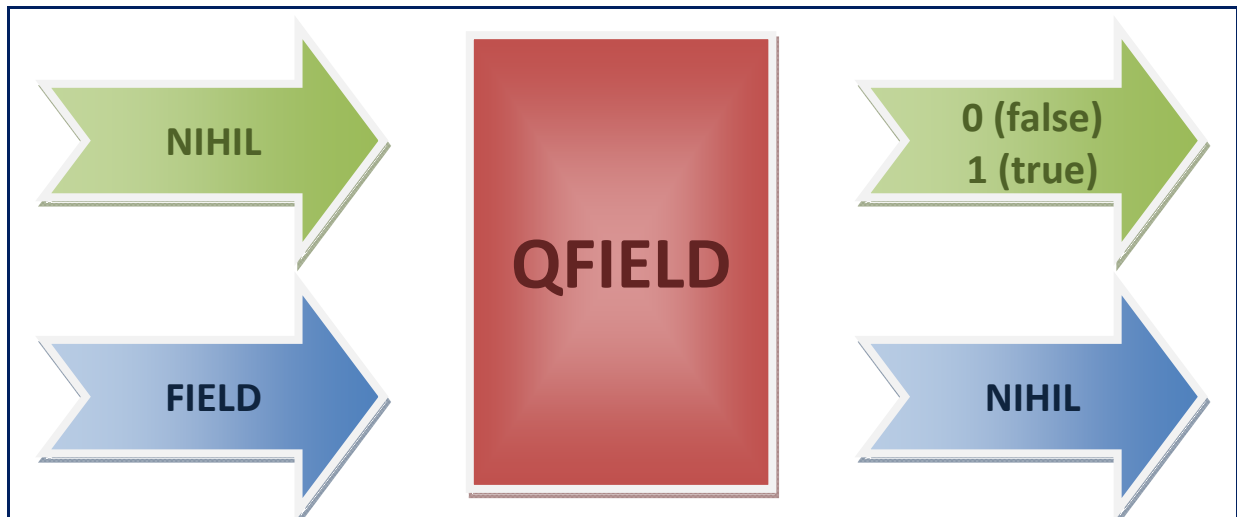


Diagram 10a: Quality Field (uitgebreide check)

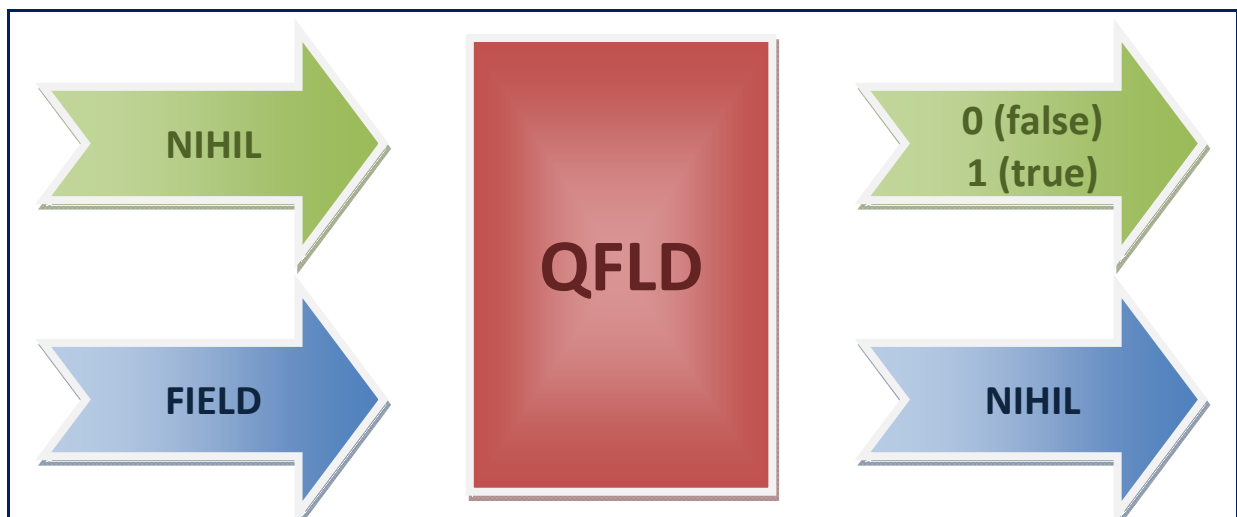


Diagram 10b: Quality field (fast check)



Het Menu

3. MENU

Dit deel beschrijft de volledige menu. Een menu zorgt ervoor dat alle programmadelen aan elkaar worden gekoppeld zodat je dit spel gemakkelijk en vlot kunt spelen.

De bestanden eindigend op **.M** zijn menu-delen (vandaar er geen « » aanwezig zijn, maar wel { }).

0. TRIO

Beschrijving:

Met dit programmadeel wordt het spel opgestart. Er wordt doorgeschakeld naar het primaire niveau.

Listing:

'TRIO'

```
« CLEAR CLEAR
"      _TRIO_v2.3_"
2 DISP
"      New_game ► _NEW"
4 DISP
"      _Cont._curr. ► _CONT"
5 DISP
"      Extra's ► _XTRA"
6 DISP 3 FREEZE P.M
TMENU
»
```

TRIO = 159 bytes

04 April 2020

1. Primair niveau Menu

Beschrijving:

De aanroep van dit menu-deel gebeurt in **TRIO**. In dit menu-deel kiest men tussen **NEW**, **CONT XTRA** of **QUIT**. Bij **NEW** gaat men verder naar **S1.M**, **CONT** gaat verder bij **T1.M**. **XTRA** gaat verder bij **C1.M** en **QUIT** verlaat het programma (en gaat naar rekenmodus).

Listing:

'P.M'

```
{ { "NEW"
« CLLCD
"_____NEW" 2
DISP
"___Level_>_3_2_1_0"
4 DISP
"___Return_>_BACK"
5 DISP 3 FREEZE
S1.M TMENU
» } { "CONT"
« CLLCD
"_____CONTINUE_GAME"
2 DISP CHKFLD
MNFRQTN MXFRQTN
  IF min 0 ==
  THEN
"The_#_indicator<
  _remains_on_0"
MSGBOX
  END
  IF 9 FC?
  THEN
"___NEXT_TN_>_→<
PREVIOUS_TN_>_←"
MSGBOX
  END
  IF pntr 0 < pntr
51 ≥ OR
```

```
THEN 0 'pntr' STO
END NXTTN SCREEN
T1.M TMENU
» } { "XTRA"
« CLLCD
"_____EXTRA's" 2
DISP
"__Reset_field_>_FIELD"
4 DISP
"__Reset__TN's_>_TN"
5 DISP
"__Quality_FLD_>_QUAL"
6 DISP 3 FREEZE
S3.M TMENU
» } { } { } {
"QUIT"
« CLLCD 9
  IF FC?
  THEN
```

```
"_____TRIO_v2.3 _____"
2 DISP 3 FREEZE
"__Thanks_for<
_____playing"
MSGBOX
  END 2 MENU
» } }
```

P.M = 714 bytes

28 May 2020

2. Secundair niveau Menu

Beschrijving:

Dit is de besturing van **NEW**. Van **S1.M** wordt naar **S1b.M** gegaan. **S1.M** heeft geen **OK**-toets, omdat er zeker een level (**3; 2; 1** of **0**) moet gekozen worden. Daarna kan er bevestigd worden met **OK** (in **S1b.M**). Hierna wordt er naar **T1.M** gegaan.

S3.M is de besturing van **XTRA**. Daar heeft men de keuze om **FIELD** en/of **TN** te resetten. Een speelveld valideren en de meldingen uit- of aan te zetten. Wanneer een van de acties is uitgevoerd, wordt er terug naar het opstartmenu gegaan (**TRIO**).

Listings:

'S1.M'

```
{ { "3"
« 3 CRTSPCFLD S1b.M
TMENU
» } { "2"
« 2 CRTSPCFLD S1b.M
TMENU
» } { "1"
« 1 CRTSPCFLD S1b.M
  TMENU
» } { "0"
« CLLCD
```

```
".....NOTE↵
The #_indicator↵
  _remains_on_0"
MSGBOX 0 CRTSPCFLD
S1b.M TMENU
» } { "BACK"
« TRIO
» } { } { "4"
« 4 CRTSPCFLD S1b.M
TMENU
» } }
```

S1.M = 337 bytes

04 April 2020

'S1b.M'

```
{ { "3"
« 3 CRTSPCFLD
» } { "2"
« 2 CRTSPCFLD
» } { "1"
« 1 CRTSPCFLD
» } { "0"
« CLLCD
"_____NOTE↵
The_#_indicator↵
_remains_on_0"
MSGBOX 0 CRTSPCFLD
» } { } { "OK"
```

```
«
  IF 9 FC?
    THEN "" 4 DISP ""
5 DISP "" 6 DISP
"__NEXT_TN_▶_→↵
PREVIOUS_TN_▶_↵←"
MSGBOX
"__NEXT_SOL_▶_>↵
PREVIOUS_SOL_▶_<↵
MSGBOX
  END GENTN SCREEN
T1.M TMENU
» } { "4"
« 4 CRTSPCFLD
» } }
```

S1b.M = 428 bytes

04 April 2020

'S3.M'

```
{ { "FIELD"
« RSTFLD CLLCD
"_Field_is_reset"
_and_qualified"
MSGBOX TRIO
» } { "TN"
« RSTTN CLLCD
"_TN's_are_reset"
MSGBOX TRIO
» } { "QUAL"
« CLLCD
  IF QFLD
  THEN
    "_Field_is"
    _qualified"
    ELSE
    "_Field_NOT"
```

```
_qualified"
  END MSGBOX TRIO
» } { "MSG"
« CLLCD
  IF 9 FS?C
  THEN
    "_UNIMPORTANT"
    _MESSAGES_ON"
MSGBOX
  ELSE 9 SF
    "_UNIMPORTANT"
    _MESSAGES_OFF"
MSGBOX
  END TRIO
» } { "BACK"
« TRIO
» } }
```

S3.M = 452 bytes

04 April 2020

3. Tercair niveau Menu

Beschrijving:

Met **NEW** of **CONT** gaat men naar het speelveld. **T1.M** heeft de mogelijkheid om alle doelgetallen nog eens dooreen te schudden (**S.TN**, Shuffle TN). Daarna wordt er sowieso naar **T1b.M** gegaan.

Listings:

'T1.M'

```
{ { "S.TN"
« GENTN
"____TN's_are←
____shuffled"
MSGBOX SCREEN T1b.M
TMENU
» } { "SOL"
« CHKSOLFLD
SOLSCREEN T1b.M
TMENU
```

```
» } { "←"
« PRVTN SCREEN
T1b.M TMENU
» } { "→"
« NXTTN SCREEN
T1b.M TMENU
» } { "BACK"
« TRIO
» } }
```

T1.M = 293 bytes

28 May 2020

'T1b.M'

```
{ { } { "SOL"
« CHKSOLFLD
SOLSCREEN C1.M
TMENU
» } { "←"
« PRVTN SCREEN
```

```
» } { "→"
« NXTTN SCREEN
» } { "BACK"
« TRIO
» } }
```

T1b.M = 187 bytes

04 April 2020

4. Quartair niveau Menu

Beschrijving:

Hier wordt de besturing van **SOL** (het oplossingsveld) gedaan. Dit deel wordt opgeroepen vanuit **T1b.M**.

Listings:

'C1.M'

```
{ { } { "CONT"  
« SCREEN T1b.M  
TMENU  
» } { "<"
```

C3.M = 149 bytes

```
« PRVSOL SOLSCREEN  
» } { ">"  
« NXTSOL SOLSCREEN  
» } }
```

04 April 2020