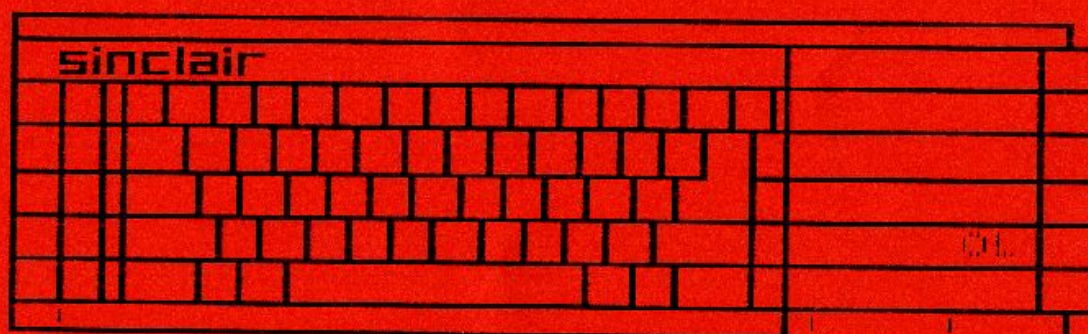


# QLONE

QL GROEP NOORD NEDERLAND



april 1988  
Jaargang 3 nummer 4



Een uitgave van : QL GROEP NOORD-NEDERLAND

VOORZITTER : J. Doesburg 2e : B. Johnson  
Kalkwijk 7 Froukemaheerd 177  
9603 BB Hoogezand 9736 RR Groningen  
tel. 05980-27633 050-418699

SECRETARIS : R. Russchen  
Westermaad 55  
8431 TB Oosterwolde  
tel. 05160-5922

PENNINGMEESTER : B. Scheidema  
Rijksstraatweg 313  
9752 CE Haren  
tel. 050-345123

REDAKTIE QLONE : J. Doesburg - A. Stam - J. van Bruggen  
Bosweg 27  
9414 BD Hooghalen  
tel. 05939-501

---

BANKREKENING : 48.09.90.468 t.n.v. B. Scheidema Haren  
GIROREKENING : 3801095 t.n.v. B. Scheidema Haren

Contributie fl. 75,--. per jaar Abonnement QLONE fl.20,--.  
Jeugdleden fl. 37,50. per jaar

---

De volgende clubavond is op 5 april 1988 in het Denksport Centrum, Oliemuldersweg 43 te Groningen. Aanvang 19.30 uur.

---

Copy voor de 15e van de maand inleveren bij, of toezenden aan de redactie. Inleveren tijdens de clubavond kan ook. Listings tekeningen e.d. (brieven als Quill document) uitsluitend op een microdrive cartridge of 3 1/2 inch floppy disk.

---

Qlone wordt gemaakt op een QL met gebruik van Quill, Archive, SuperBASIC programma's en een QL printer.

---



## INHOUD

|                        |    |
|------------------------|----|
| Van de redactie .....  | 3  |
| Toekomst II .....      | 4  |
| Super BASIC .....      | 4  |
| XINPUT .....           | 5  |
| Super BASIC .....      | 7  |
| Kretologiemixer .....  | 8  |
| Nog een keer TRA ..... | 9  |
| Cursus .....           | 9  |
| Binaire zoeken .....   | 10 |
| De muis .....          | 11 |
| Tips .....             | 14 |

## Van de redactie

Het is er dan toch van gekomen. Had ik altijd gedacht dat het virus aan de QL-gebruiker voorbij zou gaan. Maar dit keer was het dan toch raak. Het begon met een lichte hapering van de IO channels en daarna ging het systeem vrijwel down.



En zo wordt deze QLONE dan samengesteld met ter rechterzijde van mijn QL een grote stapel papieren zakdoekjes!  
By the way: nog de groeten van Hans Reinerie (en z'n QL!).

## TE KOOP

5 1/4 inch diskdrive  
ds 40 tracks  
inclusief kast en voeding

inlichtingen: J.D. Baas  
Rottummerplaat 44  
9931 ED Delfzijl  
05960 17308

## Toekomst II

Ja!. Dat is in het kort mijn antwoord op de vraag die je de vorige week stelde Bob. Maar dan zal er wel wat moeten gebeuren. Want de toekomst krijg je niet zomaar. Daar moet je wel wat voor doen!

De QLGNN is ontstaan in de pioniertijd van de QL en kon toen drijven op de verbazing van hun prille eigenaars. Maar onvermijdelijk komt er een tijd dat het nieuwe van de machine af is. Dan moet blijken of de club meer is dan een groepje (grote) kinderen dat toevallig enthousiast was over hetzelfde speeltje.

Is de QL Groep Noord Nederland volwassen genoeg om over de grenzen van het zwarte kastje heen te kijken? Te kijken in bijvoorbeeld de richting van:

- het ontwikkelen van software in groepsverband
- bestuderen van het operating systeem
- stimuleren van de interesse voor informatica in 't algemeen
- werken met machinecode
- het uitvoeren van hardware projectjes
- het leggen van contacten met andere computergebruikers

Kortom maak van de QLGNN meer een groep van computergebruikers. Want al komt er een QDOS kaart of misschien zelfs een Futura, ook dan zul je steeds weer dat moment bereiken dat het nieuwe er af is en dat men zich de cruciale vraag gaat stellen: "en wat nu". De komst van andere software kan volgens mij niet helpen. Zeg nou zelf, wat moet ik met drie tekstverwerkers, vijf spreedsheets en het zoveelste spelletje?

Dus als het aan mij ligt, op naar een QLGNN die zich richt op computers en informatica in z'n algemeenheid. Met natuurlijk de QL en QL-compatibles als instrument om die computers en informatica te bestuderen!

---

```
100 REMark dit programma werkt in tv-mode 4
110 CLS : CLS #0
120 a=2:b$='→'
130 REPEAT loop
140 FOR n=0 TO 36
150 AT 0,n*a:PRINT b$
160 SCROLL n/2
170 END FOR n
180 END REPEAT loop
```

Super BASIC

## EEN NIEUW INPUT-COMMANDO VOOR DE QL

*Deze maand begint Albert Stam met een serie artikellen over een nieuw commando voor QDOS waar velen van gedroomd hebben. Het zal de jaloeerse blikken naar Archive doen verdwijnen. Een nieuw invoer commando die default waarden kan verwerken en uitbreidingen kent wat betreft de invoerbevestiging. Deze maand een inleiding over het hoe en waarom.*

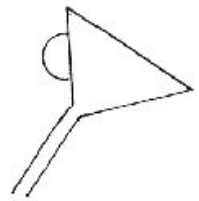
De gebruikersvriendelijkheid van een programma is vaak afhankelijk van de manier waarop gegevens ingevoerd kunnen worden. Het is duidelijk dat zodra meerdere gegevens ingetikt moeten worden waarbij de manier van invoer aan te veel regels gebonden is, de toegankelijkheid van het programma flink wordt beperkt. Een goede illustratie van het tegenovergestelde is Archive: de invoer is vrijwel niet aan regels gebonden, men kan vrijelijk velden invullen onafhankelijk van enige volgorde. Maar ook als de programmeur het programma te algemeen wil houden, dus zo weinig mogelijk regels wil gebruiken waaraan de gegevens moeten voldoen om het in het programma te laten passen, dan gaat dat ten koste van de flexibiliteit van de manier waarop de gegevens ingevoerd mogen worden.

Als men bijvoorbeeld een hoeveelheid records moet invoeren en men moet steeds opnieuw aangeven welk veld men wil invullen en hoeveel ruimte men daarvoor nodig denkt te hebben dan kan dat tot flinke frustraties leiden bij de gebruiker. Terwijl het programma heel goed de algemeenheid kan dienen door geen onderscheid te maken (vergelijk: geen regels te kennen) tussen bijvoorbeeld numerieke en alfanumerieke gegevens of b.v. geen maximale lengte kent.

Ook hier illustreert Archive min of meer het tegenovergestelde: het maakt onderscheid tussen letters en getallen en heeft een maximale gegevenslengte terwijl Archive toch een enorm flexibele invoermethode kent.

Wat ook de toegankelijkheid bekort is als een waarde die voor de hand ligt steeds opnieuw ingevoerd moet worden. Dit is ook een beperking van Archive. Als men bijvoorbeeld een telefoonnummer-bestand, het aloude voorbeeld, wil opzetten en men wil het telefoonnummer-veld zo algemeen mogelijk uitvoeren door ook de telefoonnummers buiten de regio in hetzelfde veld te kunnen plaatsen. Hierdoor is het dus nodig ook de kengetallen te vermelden. Maar de gebruiker heeft nogal veel kennis in de eigen regio dus veel dezelfde kentallen. Handig zou zijn als het programma zelf het eigen kental alvast adviseert en de gebruiker alleen maar aan te geven of hij met deze waarde eens is, zo niet moet hij gewoon een ander kengetal invoeren zonder dat het programma z'n voorhoofd hoeft te fronsen. Dit noemt men een defaultwaarde.

Als alle programma's op deze basis-voorwaarden om: "need more, more input, INPUT!" gaan vragen dan zijn in ieder geval de nachtmerries van de datatypistes verleden tijd. Het (sterke) punt van een computerprogramma is echter dat elk programma weer anders is.



Dus is het nodig dat deze basis-voorwaarden ook op de juiste manier gebruikt worden en bijvoorbeeld geen onzinnige default-waarden adviseert. Het is nog steeds nodig dat de programmeur met de gebruiker meedenkt en van te voren bedenkt welke situaties (lees: gegevens) vaak voorkomen en daarnaar zijn invoermethodes ontwerpt.

Tot zover de theorie. Maar hoe bouw je nu zo iets? Of, om mee te beginnen, aan welke regels moet de methode voldoen? Het moet flexibel genoeg zijn en toch superalgemeen toepasbaar want we willen natuurlijk niet dat we bij elk programma een nieuwe invoermethode moeten ontwerpen. Het is dus nodig dat we het probleem zo fundamenteel mogelijk aanpakken. Door te beginnen bij het INPUT commando bijvoorbeeld. Met Superbasic's INPUT commando hoef je het niet te proberen. Daar kun je nog zoveel ingewikkelde procedures omheen bouwen, maar het zal je niet lukken om b.v. een default waarde mee te geven laat staan functietoetsen uitlezen. En zodra een ingewikkelde procedure wordt ontworpen met INKEY\$ of desnoods Toolkits GET of BGET, loopt de boel weer veel te langzaam. "Compileren!" hoor ik iemand roepen. Doe maar, als je minstens 640K tot je beschikking hebt. Ik vind zo iets een geforceerde oplossing en bovendien is het veel leuker zelf verder te zoeken. Een van de vele redenen waarom ik Supercharge of Qliberator niet bezit. We ontkomen er niet aan: Machinetaal (met en hoofdletter!) De voordelen. De snelheid, de compactheid, de lol, de toepasbaarheid (niet alleen binnen Superbasic) en de uitwijkmogelijkheden om maar wat te noemen.

Wel, dan besluiten we hier dat we een nieuw commando aan de QDOS gaan geven, een naam heb ik ook al bedacht: XINPUT, een samentrekking van eXtended en INPUT.

Allereerst de specificaties. Hoe gaat het commando er uitzien. Natuurlijk moet het toepasbaar zijn in elke console window dus moeten we een channel nummer meegeven en bovendien moeten we testen of dat kanaal open is en of er inderdaad een console window aan hangt. Verder zijn we natuurlijk een variabele nodig waarin we de ingevoerde waarde kunnen plaatsen en moeten we dus weer testen of dat mogelijk is.

Vervolgens, we willen een defaultwaarde mee kunnen geven dus we zijn een derde parameter nodig die de defaultwaarde voorstelt. Meer parameters zijn niet nodig volgens mij. De cursorpositie wordt zoals bij alle Superbasic commando's door AT ingesteld. Er zouden nog extra faciliteiten toegevoegd kunnen worden zoals een byte die aangeeft welke character we moeten gebruiken als cursor of een parameter die aangeeft of het een knipperende cursor (ook in mode 4) moet zijn of niet.

Een parameter die toestaat maar een bepaald aantal characters in te toetsen. En waarschijnlijk valt er nog veel meer te bedenken maar ik denk dat dat soort parameters op den duur allemaal overbodige programmeervriendelijkheid wordt en altijd op een vaste waarde komen te staan. Dus dat soort opties kunnen het best van te voren worden uitgezocht om vervolgens de beste mogelijkheid in te programmeren.

Dan moet het verder mogelijk zijn om de functietoetsen en de ESC en TAB in- of exclusief SHIFT, ALT en/of CTRL uit te lezen zodat die door de programmeur een aparte functie kunnen worden toebedeeld zoals: -Terug naar het vorige veld, of -Wis dit hele record enz. Omdat deze mogelijkheden in geen enkele QDOS rom-routine aan bod komen zullen we zelf een totaal nieuwe INPUT moeten schrijven en daarom ook rekening moeten houden met: de cursor loopt van de regel af, er moet een karakter gewist worden, de cursor moet naar een andere positie verplaatst worden, de cursor kan niet verder verplaatst worden omdat het aan het eind staat enzovoort, enzovoort.

Tot zover deze inleiding, volgende maand begin ik met het algoritme. Ook zal dan de BASIC versie van XINPUT worden geplaatst om het algoritme toe te lichten en de compileer-gekken tegemoet te komen.

Albert Stam<

---

```
100 MODE 4:WINDOW 512,256,0,0:PAPER 0:CLS
110 RESTORE 230
120 DIM t$(17,35):FOR n= 1 TO 7:READ t$(n)
130 write .8
140 DEFine PROCedure write (MD)
150 WINDOW 448,210,32,16:PAPER 0:CLS:CSIZE 2,1
160 FOR i= 1 TO 7
170 FOR n= 2 TO 7*MD
180 IF (n/2=INT(n/2)) OR (n= 7 * MD):OVER 1
190 pos= (1*20+16+3*(MD))
200 INK INT(n/MD):CURSOR 16+n-3*MD,pos-n
210 PRINT t$(i):OVER -1:NEXT n:NEXT i:OVER 0
220 END DEFine write
230 DATA '
240 DATA '          CLIVE SINCLAIR
250 DATA '          presents
260 DATA '
270 DATA '          THE INCREDIBLE
280 DATA '          SINCLAIR  QUANTUM LEAP
290 DATA '          ,
```



## KRETOLOGIEMIXER

Ook een beginnende QL gebruiker moet in staat worden geacht om het onderstaande verhaal in een programmaatje te gieten.

### *hulp bij zinvolle discussie*

| KOLOM 1                   | KOLOM 2             | KOLOM 3                  |
|---------------------------|---------------------|--------------------------|
| 1. geïntegreerde          | 1. organisatorische | 1. modelontwikkeling     |
| 2. systematische          | 2. ontwikkeling van | 2. gedragalternatieven   |
| 3. functionele            | 3. benadering van   | 3. managementtechnieken  |
| 4. multi-disciplinaire    | 4. technologische   | 4. organisatievorm       |
| 5. normatieve             | 5. relationele      | 5. spanningsvelden       |
| 6. sociaal-psychologische | 6. coördinatie      | 6. machtsaspecten        |
| 7. dynamische             | 7. bureaucratische  | 7. informatiesystemen    |
| 8. geöbjectieveerde       | 8. prognose van     | 8. communicatiemethoden  |
| 9. structurele            | 9. conservatieve    | 9. startegiebepaling     |
| 0. integrale              | 0. begeleiding van  | 0. veranderingsprocessen |

Het gebruik is heel eenvoudig:

Kies een willekeurig getal van drie cijfers; ieder van deze drie cijfers correspondeert met een kolom. Neem bijvoorbeeld het getal 430:

- kies woord 4 uit kolom 1
- kies woord 3 uit kolom 2
- kies woord 0 uit kolom 3

Het resultaat is: "multi-disciplinaire benadering van veranderingsprocessen". Een uitdrukking die iedereen zal bevredigen. Het doet er niet toe dat u niet weet wat het betekent, dat weet uw gesprekspartner ook niet en hij zal er wel voor waken het te vragen.

Misschien geeft u de voorkeur aan:

"normatieve begeleiding van managementtechnieken" (503) of:  
 "geïntegreerde technologische machtsaspecten" (146).

Zeer duidelijk is ook: "geobjectieveerde rationele gedragalternatieven" (852). Op deze manier ontstaan waardevolle bijdragen voor een zinvolle discussie.

De kretologiemixer verschaft de gebruiker duizend verschillende termen die aan uw conversatie juist die klank weet te geven van autoriteit, die uitsluitend wordt voortgebracht door kennis en diep inzicht.

(van de muur gehaald op een automatiseringsafdeling)



## NOG EEN KEER TRA

Ik heb begrepen dat veel QL-ers de opbouw van de seriële tabel niet goed konden volgen in mijn artikel van februari 1988. Het blijkt dat er een fout in de beschrijving zit. Mijn excuses hiervoor. Daarom hieronder nog een maal de beschrijving van de seriële tabel voor TRA.

| rel. adres | hex    | dec   | opmerking                        |
|------------|--------|-------|----------------------------------|
| \$00.W     | \$4AFB | 19195 | nonsense waarde                  |
| \$02.W     | \$0005 | 5     | offset naar eerste tabel         |
| \$04.W     | \$0106 | 256+6 | offset naar tweede tabel         |
| \$06.B     |        |       | vertaling voor char. 0 zie onder |
| \$07.B     |        |       | vertaling voor char. 1           |
| \$08.B     |        |       | vertaling voor char. 2           |
| !          |        |       | !                                |
| \$104.B    |        |       | vertaling voor char. 254         |
| \$105.B    |        |       | vertaling voor char. 255         |
| \$106.B    |        |       | aantal extra vertalingen         |
| \$107.B    |        |       | te vertalen char.                |
| \$108.B    |        |       | 1 <sup>e</sup> vertalingsbyte    |
| \$109.B    |        |       | 2 <sup>e</sup> vertalingsbyte    |
| \$10A.B    |        |       | 3 <sup>e</sup> vertalingsbyte    |
| \$10B.B    |        |       | te vertalen char.                |
| enz.       |        |       | enz.                             |

De eerste tabel bevat de vertalingen bestaande uit één byte. B.v. de charactercode voor uw printer van de character (PI) is geen 177 (QL character set) maar (b.v.) 178. Dan plaatst u op de byte voor de vertaling van code 177 het getal 178. De characters die niet vertaald moeten worden houden hun eigen waarde. Als u een nul op de te vertalen positie plaatst wordt de tweede tabel gebruikt. Daar hebt u meer ruimte voor de vertaling, nl. 3 bytes, bent u maar twee nodig, plaats dan een nul in de derde.

## CURSUS

Bij voldoende deelname zal Bob Johnson op de computeravonden weer een cursus SuperBASIC geven. René Russchen zal, eveneens bij voldoende deelname, een cursus geven in het gebruik van de standaardpakketten: Quill, Abacus, Archive en Easel.

### Binaire zoeken

Het principe van binaire zoeken is vrijwel iedereen bekend. Velen zullen het zelfs al eens (intuïtief) gebruikt hebben. Een voorbeeld: Een artikel kost 125 gulden. Een kandidaat in een quiz moet de prijs van dat artikel raden. Hij weet dat de prijs in hele gulden tussen 0 en 200 gulden moet liggen. Onze kandidaat is systematisch van aard en begint op de helft van 0 en 200, 100 gulden dus. Dat blijkt te laag. Zijn volgende gok ligt dan midden tussen 100 en 200, op 150. Te hoog. Vervolgens gaat onze kandidaat precies tussen 100 en 150 gulden, dus op 125 gulden zitten. Bingo!

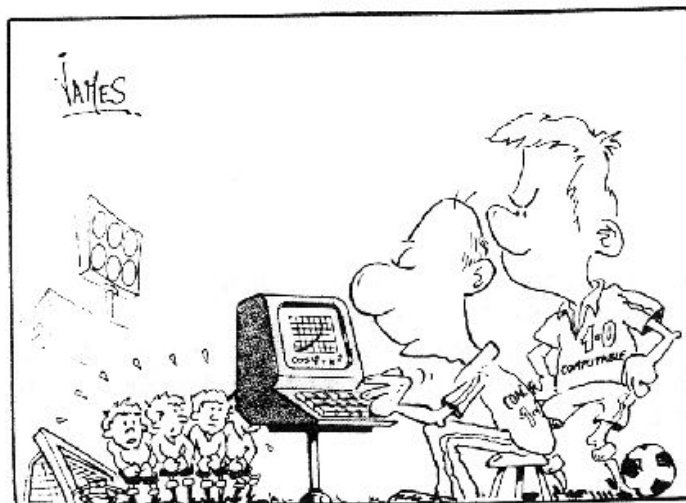
Dat is dus binaire zoeken. De bruikbaarheid van dit principe strekt zich echter tot ver buiten het kwisgebeuren uit!

In iedere tabel die op het zoekargument gesorteerd is, is binaire zoeken mogelijk. Voor het doorzoeken van kleine hoeveelheden gegevens is de methode van binaire zoeken "ver gezocht". Moet je daarentegen grotere hoeveelheden informatie doorzoeken, dan biedt de binaire zoekmethode het grote voordeel dat het aantal zoekslagen relatief langzaam toeneemt t.o.v. de hoeveelheid te doorzoeken informatie.

Het maximale aantal zoekslagen is namelijk gelijk aan de (grootste macht van 2) +1 in het aantal van de te doorzoeken gegevens. Voorbeeld:

1000 records gesorteerd opgeslagen  
grootste macht van 2 in 1000 = 9  
maximaal aantal zoekslagen = 9+1 = 10

Rond het principe van binaire zoeken laat zich natuurlijk uitstekend een Super BASIC zoekroutine schrijven. Maar zoek dat zelf maar een uit!





**MUIS VINDT KORTSTE WEG IN DOOLHOF.**

Een muis wordt op een bepaalde plaats in een doolhof gezet. Ze gaat op zoek naar de uitgang. Al gauw wordt het duidelijk dat ze niet dom is. Ze zoekt systematisch de weg. Op een kruising aangekomen slaat ze af in één van de paden die erop uitkomen. Is het een doodlopend pad dan gaat ze door tot het eind - wel een beetje dom dus - en daarna snel weer terug naar de kruising. Ze neemt dan het volgende pad. Loopt ook dit pad dood dan gaat ze weer door tot het eind - ze is dus nog steeds een beetje dom - en daarna snel weer terug. Terug op de kruising blijkt dat ze van al haar zoeken heeft geleerd. Er is nog één pad dat ze nog niet heeft onderzocht. Die slaat ze in. Ze is zo verstandig om het pad vanwaar ze de eerste keer op de kruising is aangekomen niet weer in te slaan en ook de onderzochte doodlopende paden slaat ze over. Elke volgende kruising of T-splitsing onderzoekt ze precies zo. Op deze manier komt ze feilloos bij de uitgang. Wordt ze weer teruggezet op de plaats vanwaar ze haar speurtocht is begonnen dan zien we met verbijstering dat ze zonder aarzelen snel de kortste weg neemt naar de uitgang. Ze heeft dus van haar speurtocht geleerd en de kortste weg in haar geheugen geprent. Een intelligent beestje!

Hoe je de gedragingen van de muis in SUPERBASIC simuleert is te lezen in de vijfde aflevering van de serie "Superbasic" in "QL World" van februari 1988. Ik vond het zo'n slim beestje dat ik de aanpak voor het opzetten van het programma én het programma zelf heb doorgespit. De programmeertrucjes die worden toegepast zijn erg efficiënt. De serie in "QL World" - die overigens nog loopt - wil de lezer bijbrengen op welke wijze gecompliceerde problemen kunnen worden aangepakt. Voorop staat dat men in zo'n geval niet direct SUPERBASIC-code gaat intikken maar eerst in woorden de logische gang van zaken weergeeft.

**DOEL.**

Simuleer in een SUPERBASIC-programma de gedragingen van de muis.

**SPECIFICATIE.****- Doolhof.**

Beschouw het doolhof opgebouwd uit rechthoeken. Elke rechthoek kan deel uitmaken van een pad of een muur. De muis moet bij haar speurtocht onderscheid kunnen maken tussen muur en pad.

**- Muis.**

1. Ga stap voor stap door tot de uitgang is gevonden of het einde van een doodlopend pad is bereikt.
2. Is het einde van een doodlopend pad bereikt ga dan terug naar de kruising/T-splitsing die het laatst is onderzocht.
3. Onderzoek een ingeslagen richting op een kruising/T-splitsing maar één keer.
4. Na beëindiging van de speurtocht moet de kortste weg van het vertrekpunt tot de uitgang in het geheugen vast liggen.

Als de muis zich aan de regels 1 t/m 3 houdt zal ze zonder mankeren de uitgang vinden. Als ze bovendien de lokaties die ze bij haar terugtocht van het einde van een doodlopend pad voor de tweede keer bezoekt uit haar geheugen wist zal uiteindelijk de kortste weg naar de uitgang in haar geheugen vast liggen. Voorwaarde is evenwel dat in het doolhof geen paden zijn die in een kringetje lopen.

De volgende stap is het in woorden weergeven hoe men het programma in SuperBasic denkt te schrijven.

De lokaties in het doolhof die de muis op haar speurtocht onderzoekt evenals de "illegale" richtingen van die lokaties, worden op een stack gezet. Illegale richtingen zijn die richtingen die door de muur zijn geblokkeerd, de richting die de muis vanaf de betreffende lokatie inslaat naar een volgende lokatie en de richting vanwaar de muis een lokatie bereikt.

Regel 1 wordt:

- Haal achtereenvolgens de lokatie waar de muis zich bevindt en de illegale richtingen van die lokatie van de stack.
- Controleer of een "legale" richting kan worden ingeslagen.
- Alvorens de muis de volgende stap in de legale richting maakt, worden de illegale richtingen én de gevonden legale richting - die immers nu illegaal is - evenals de lokatie waar de muis zich bevindt weer op de stack teruggezet.
- Nu pas neemt de muis de stap naar de volgende lokatie.
- Controleer ook van deze de illegale richtingen en zet deze samen met de richting vanwaar de muis de lokatie heeft bereikt én de lokatie zelf op de stack. Zet deze regels in een loop.

Als de muis het eind van een doodlopend pad heeft bereikt en weer teruggaat wordt regel 2:

- Verwijder de lokatie dat het laatst is bezocht en de illegale richtingen van die lokatie van de stack.
- Ga terug naar de laatst bezochte lokatie.
- Als de lokatie een kruising/T-splitsing is, ga dan na of er een nog niet onderzochte richting is.
- Zo ja, sla deze in. Zo nee, ga verder terug.

#### **ANALYSE VAN HET PROGRAMMA.**

Om niet te veel beslag te leggen op de ruimte in "QLONE" wordt voor de listing van het programma verwezen naar "QL World" van februari 1988.

Als de werking van de programmeertrucjes duidelijk is spreekt het programma voor zich. Van belang zijn:

- de plaats waar de muis zich bevindt.
- de "illegale" richtingen.
- de stackpointer.



De plaats waar de muis zich bevindt.

De plaats waar de muis zich tijdens haar speurtocht bevindt wordt door het programma gecontroleerd. De x- en de y-coördinaat die hiervoor nodig zijn worden in één enkel getal ondergebracht (de variabele "Here"). "Here" is de som van een veelvoud van "Size" en een fractie van "Size". Het aantal keren dat "Size" op "Here" deelbaar is, is de x-coördinaat; de rest van de deling is de y-coördinaat. (Procedure Show\_mouse, Function Analyse).

De "legale" en "illegale" richtingen.

Elke lokatie dat deel uitmaakt van het pad heeft aangrenzende elementen waaraan alleen dan, als het element deel uitmaakt van het pad, een getal wordt toegekend, te weten:

- 1 voor het element links ervan,
- 2 voor het element erboven,
- 4 voor het element eronder en
- 8 voor het element rechts ervan.

NIET GEDACHT HÈ,  
DAT WIJ ZÓ POPULAIR  
ZOULDEN WORDEN.

1, 2, 4 en 8 zijn machten van 2;  $2^0$ ,  $2^1$ ,  $2^2$  en  $2^3$ . Hiervan wordt in het programma op een efficiënte en bijzonder fraaie wijze gebruik gemaakt. Met Function Analyse worden de "legale" richtingen gevonden. 15 - Analyse geeft de "illegale" richtingen.



De muis start haar onderzoek.

(regel 906) De startwaarde van "Illegal" wordt van stack gehaald. (908) Bij de eerste doorloop van "Repeat Loop" - Prt=0 - wordt "Illegal" berekend. (910) Illegal=0. (912) De "legale" richtingen worden berekend. (914) Onderzocht wordt achtereenvolgens of links, boven, onder en rechts van de lokatie waar de muis zich bevindt een "legale" richting is. (914) Zodra een "legale" richting wordt gevonden wordt deze ingeslagen. (918) "Illegal" samen met de ingeslagen richting - die immers nu ook "illegaal" is wordt op stack gezet. (920-928) "Here" de nieuwe lokatie wordt berekend. (934) De "legale" richtingen van de nieuwe lokatie worden berekend. (936) De "illegale" richtingen samen met de richting vanwaar de nieuwe lokatie is bereikt wordt op stack gezet.

De muis bereikt het einde van een doodlopend pad.

De lokatie op het eind van een doodlopend pad heeft aan drie zijden een muur ofwel drie "illegale" richtingen, (936) die samen met de richting vanwaar de muis op die lokatie is gekomen op stack wordt gezet. Illegal=15 (906) "Illegal" wordt weer van stack gehaald. De pointer "Ptr" wordt met twee verlaagd. (910) De IF-loop wordt genegeerd. Immers Illegal=15. Dit gaat net zo lang door tot een lokatie wordt gevonden met een "legale" richting etc. tot (932) de uitgang wordt gevonden. De kortste weg staat nu keurig op stack en wel van de top tot waar de pointer wijst.

Het is nu niet moeilijk een Procedure te schrijven waarmee de intelligentie van de muis kan worden aangetoond.

## TIP(S)

Mensen die met lange tot zéér lange SuperBASIC programma's werken zouden eens moeten overwegen de QLOAD en QSAVE utilities te gebruiken. Qsave zet je programma niet weg als ASCII maar als tokens. Dat wil zeggen, één code per statement en dat scheelt behoorlijk bij het laden van het programma.

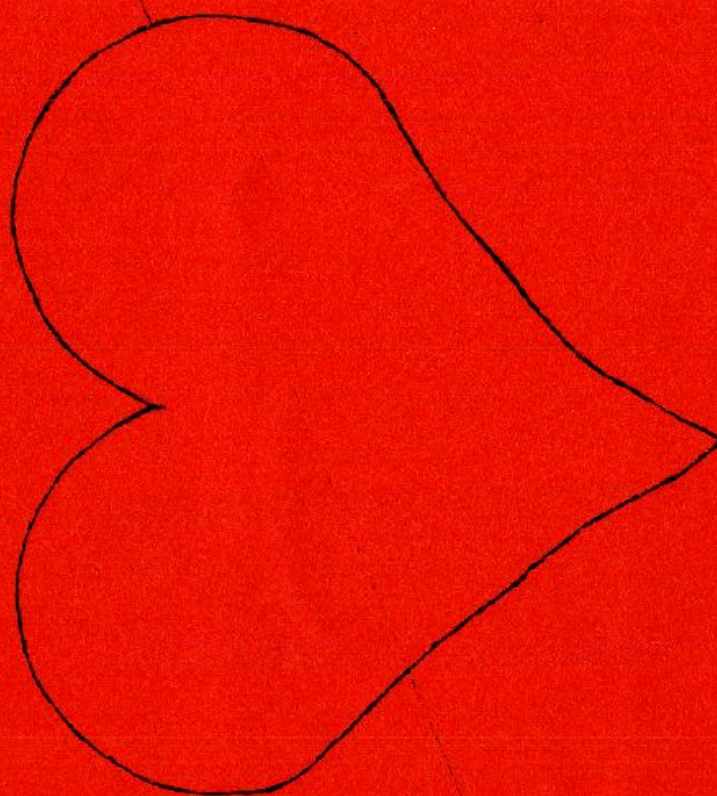
Was ik nou de enige die niet wist dat onder de SHIFT F5 van QUILL de functie "redraw screen" zat?



QL screendump



7  
FIRST  
AT FIVE  
BYTE



QLONE 70



**DRUKWERK**

Afz. Ql.G.M.M.  
Redaktieadres  
Bosweg 27  
9414 BD HOOGHALDEN



Krimpen K van  
Koldaker 34  
9407 BM Assen

113.01