



table des noms internes (initialisé)

tables fixes, non initialisées
(index table de noms...)

help

library

} source

tampon source

flag de non destruction de la source

pile de gosub, proc, fonction

local

pile de for

exec [développement de instructions]

conditions

table des mem: files

table des random- files

table de noms/index

tampon

variable
chaînes

adresses des
chaînes/variables

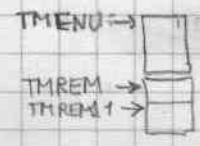
pile
utilisateur

pile interne
variables
et/ou internes

description de
chaînes/variables

pile_proc

pile_exec

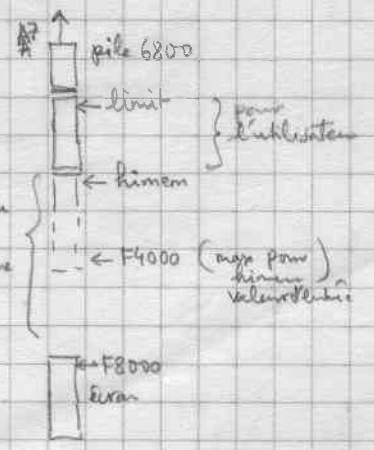


menu

remember

tampon-menu

pile-rem



AF pile 68000

limit USER

alloc MEM

F2000 écran

alloc

pile } gosub
proc
function

00 arrêt

→ valeur précédente de TPILE

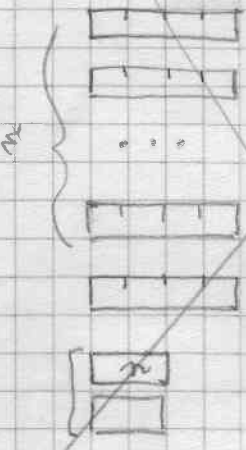
→ valeur de TVARN

(TVARN en début d'instruction)

valeurs de début d'instruction

variable attribuée à la fonction

(= TVARN au moment de l'appel la variable est initialisée lors de l'appel)

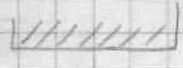


pointe 1er argument
" 2e "
" n^{eme} "
" après l'appel

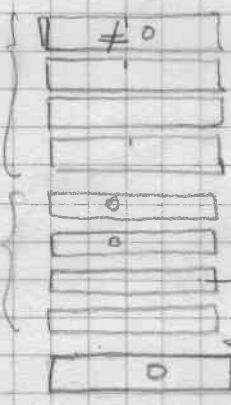
nb d'argument
0 arrêt
1 procédure
2 fonction variable
3 fonction chaine

TMPROC → ligne

pile for
TMPROC



↑ TMFOR



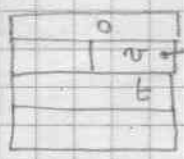
pas DS
max/min D4
adresse de l'index A3
1er instruction de boucle AS

} boucle for...next

0 while...wend
1 do...loop
2 repeat...until

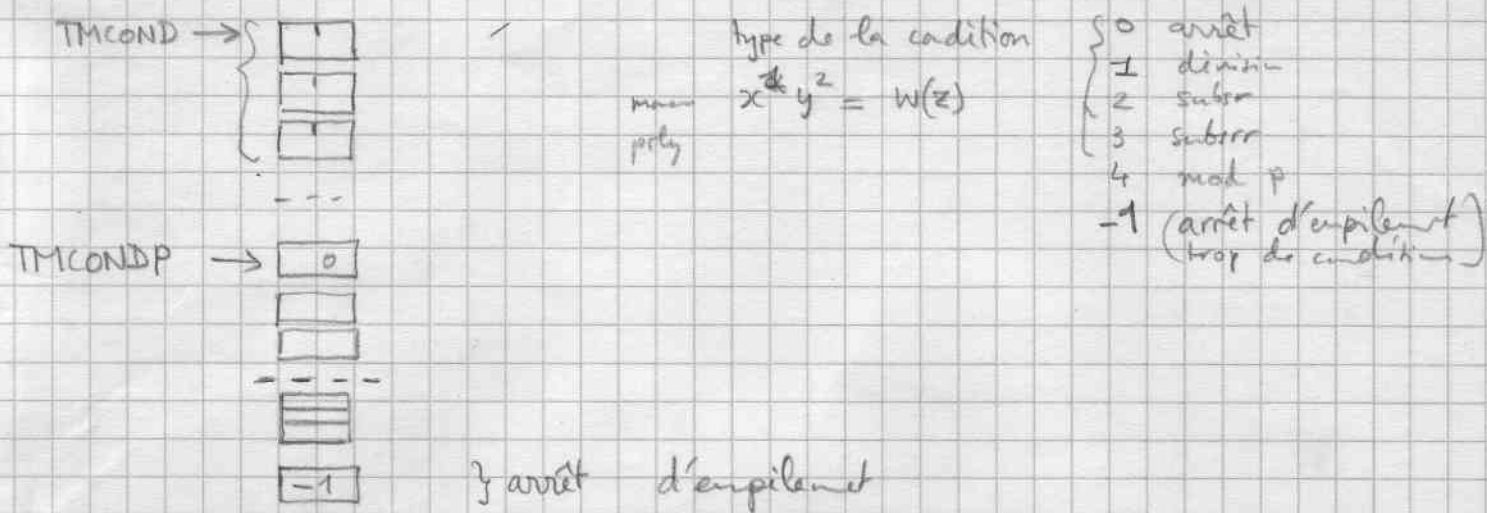
adresse de reprise (après pour DO/REPEAT ; sur expr pour WHILE loop)
arrêt

fin TMEXC

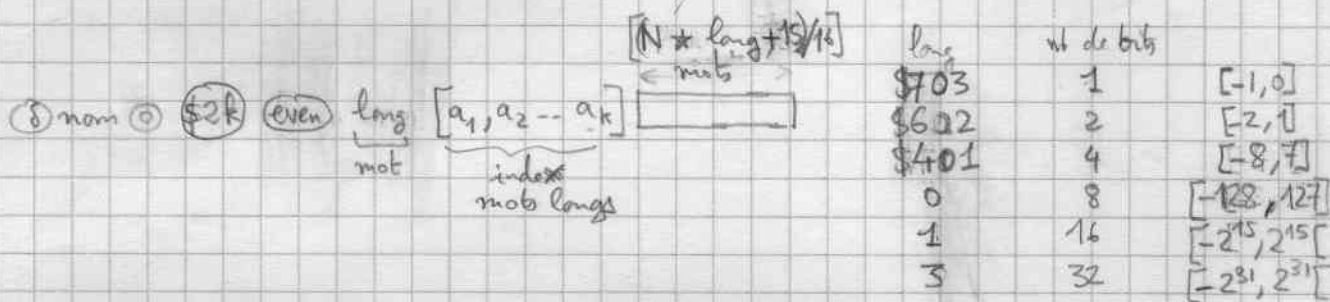


variable chaine (numéro)
3 4
adresse après forv forc

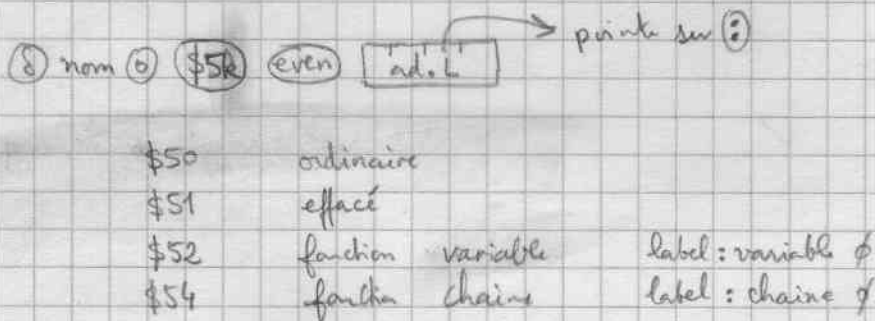
pile des conditions



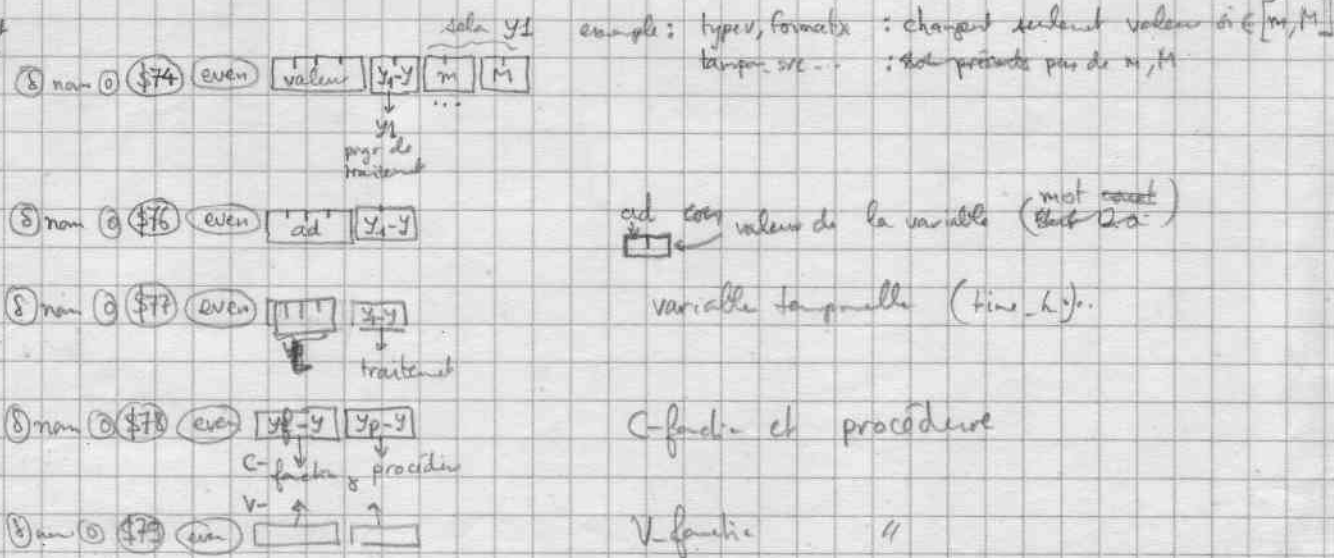
index



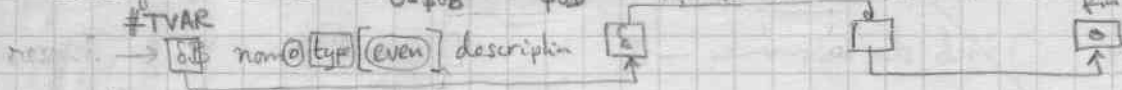
label



variable d'état



Codage des noms (fonction, commande, littéral, index, variable, string, label)



index [0] nom [0] [20] even, [10] W valeur de l'index Index: mot signé (comme les exposants)

var [0] nom [0] [10] even, [10] W le numéro est m

var [0] lit (x1...xk) [10] nom [0] [10] even, [10] W [a1W... akW] le numéro est m - d1 - d2a1 - d3a1a2 - ... - dk a1 a2 - ak-1

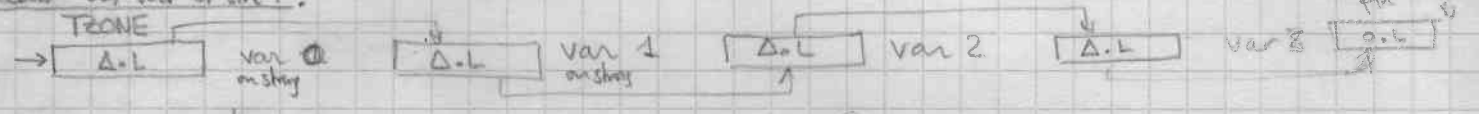
2 numérotations [A] variable et string de 0 à 7FFF
[B] littéral de 1 à 7FFF

Número suivant litu = m + 1

fonctions [0] nom [0] [10] even [10] W [data] renvoi en WSUBS + r [0]

- 0 fonction sans argument
- 1 f(expr) } expr exacte
- 2 f(expr, [])
- 3 f(poly)
- 4 f(poly, [])
- 5 f(real)
- 6 f(real, [])
- 7 f(W) } numéros flottant réel ou complexe Set r non utilisés
- 8 f(W)
- 9 f(W)
- 10 f(chain)
- 11 f(chain)
- 12 commande de branchement } illegales en mode direct
- 13
- 14
- 15

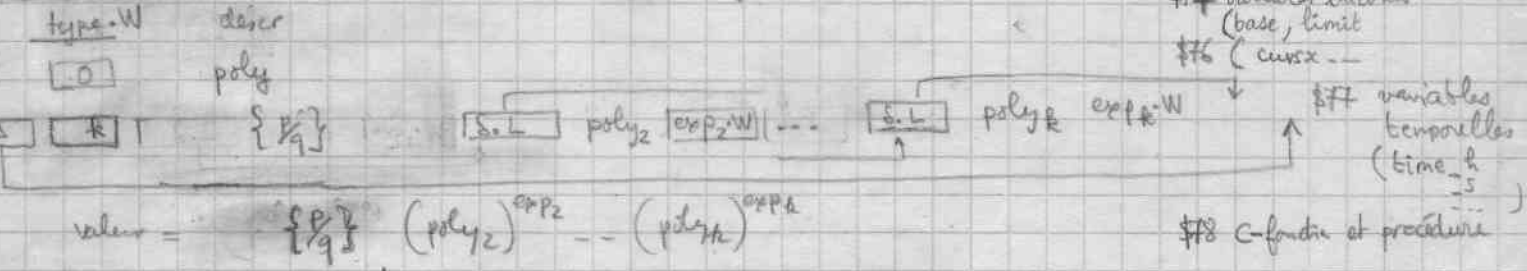
Label [0] nom [0] [50] even [10] L adol adresse valeur des var et string



- Type AND \$Fo
- \$00 - \$F interne (fact, commande) } numérotés de 1 à TVARL-1
 - \$10 - \$1F littéral } valeurs en m
 - \$20 index [D.W] } numérotés de 0 à TVARV-1
 - \$30 - \$3F var
 - \$40 - \$4F string
 - \$50 label
 - \$60 - \$6F fonction string

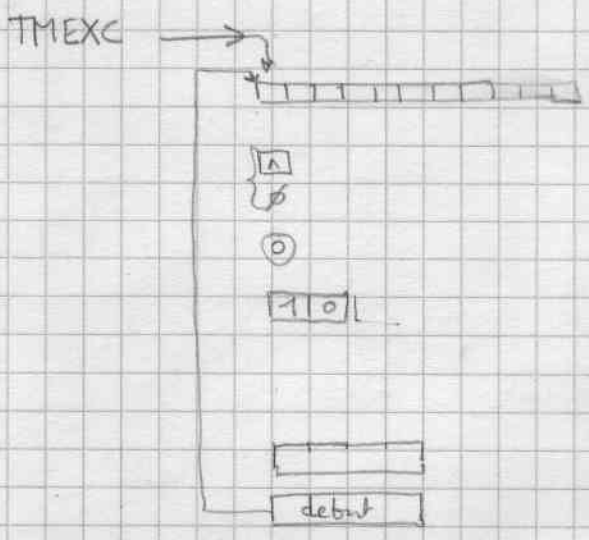
Descripteur des valeurs des string:

Descripteur des valeurs des var:



- \$70 constantes: [0] nom [0] \$70 even [0]
- \$71 "then"
- \$72 "of" "data" "data" "data"
- \$75 {xor or and
- \$76 value (dans)
- \$77 variables internes (base, limit, cursx)
- \$78 variables temporelles (time_h, ...)

pile exec



instr

odd

← permet de reconnaître que l'on est dans la pile exec

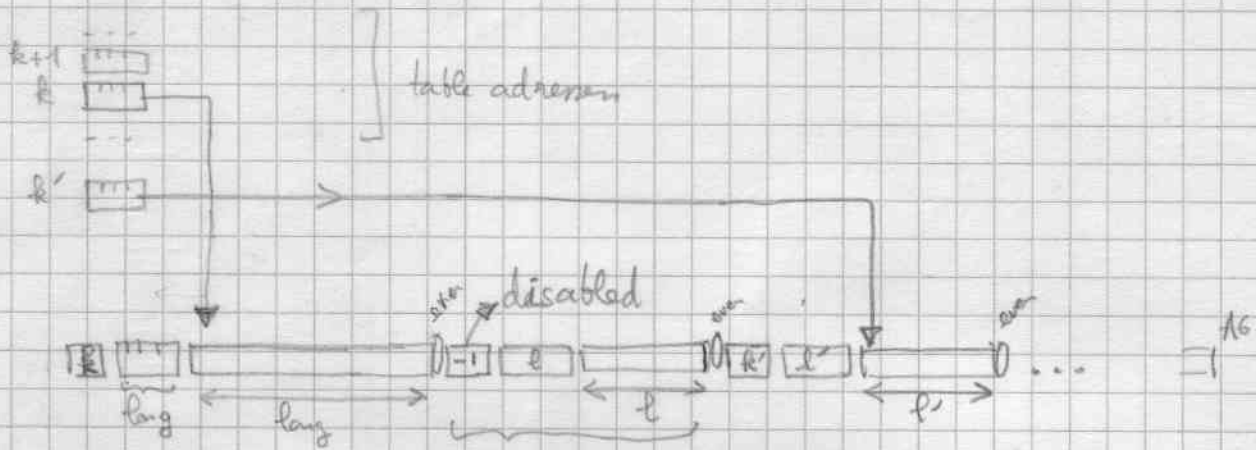
AS : début de l'instr suivante
permet le dépilege

TMEXCP →

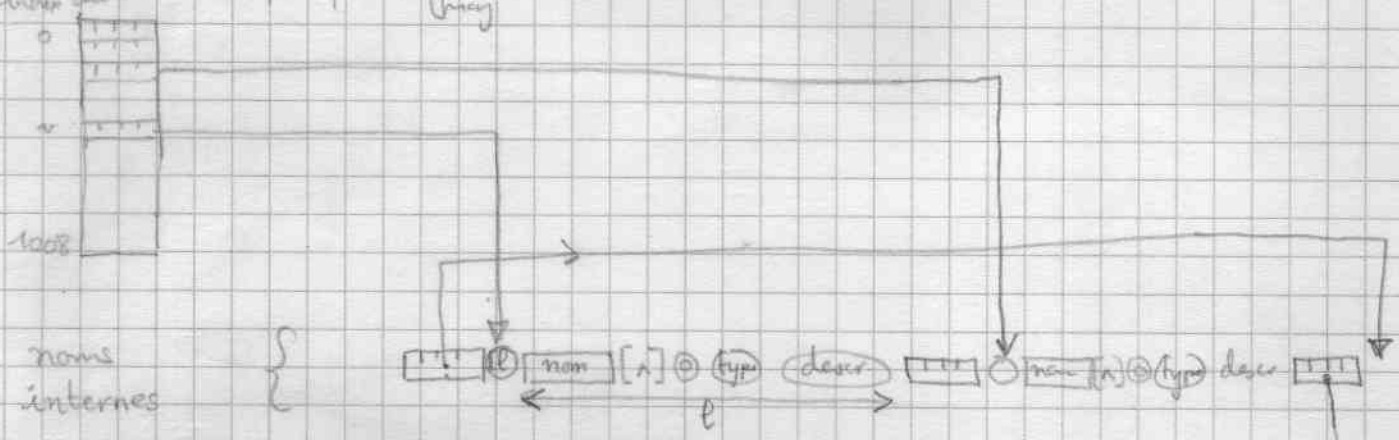
TMCOND → // // // //

valeurs max

variable / chaîne : description

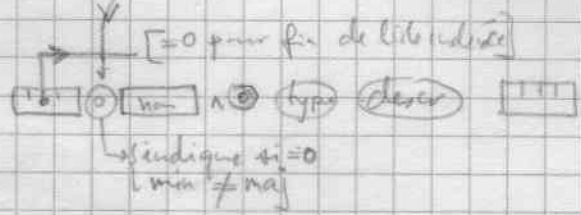


nom
L'index des nom se déduit par def min / max



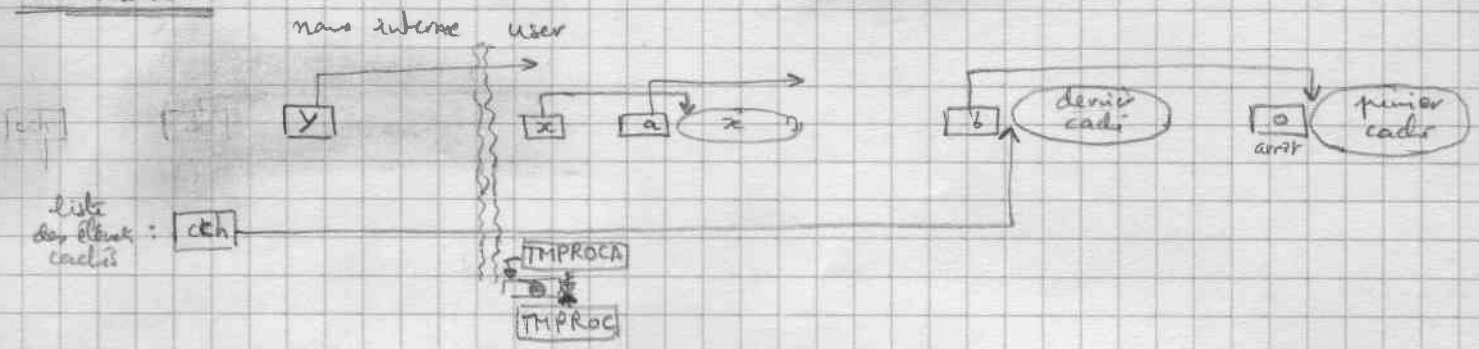
nom internes

nom externe



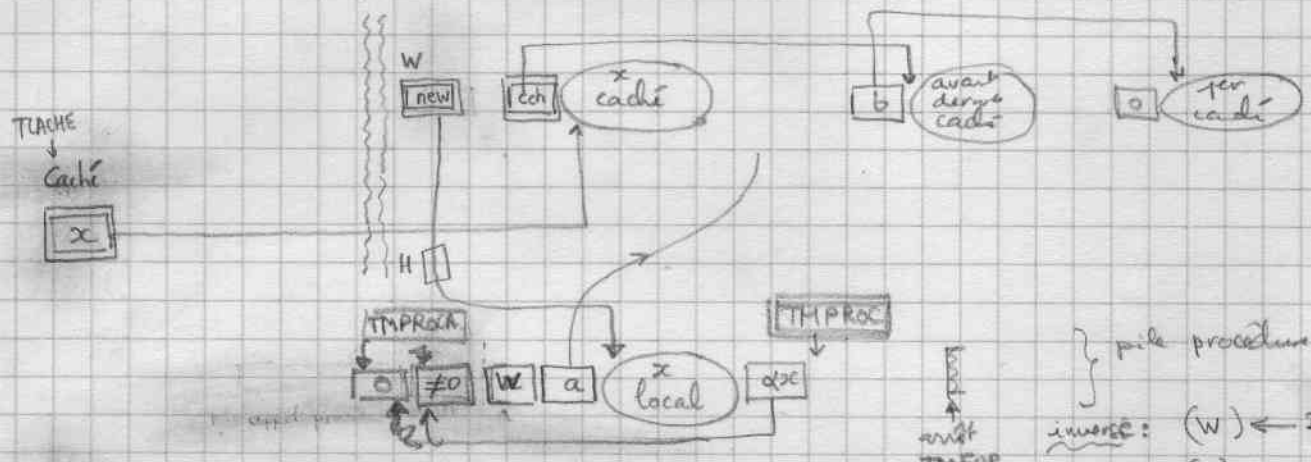
variables chainées } locales et cachées
 - lit
 - index

Types des noms:



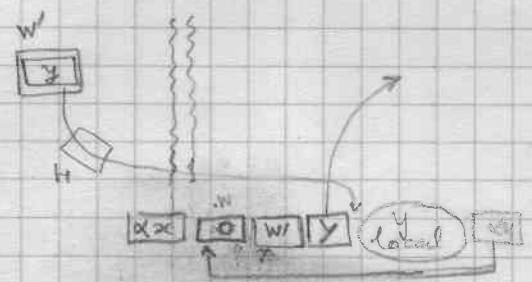
Local (variable) x

modifié ainsi: (☐ = modifier)



Local (var) y

(pas d'y précédent)



} pile procedure
 inverse: (W) ← x cache
 (x) ← a
 (cache) ← cch

inverse: (W') ← y

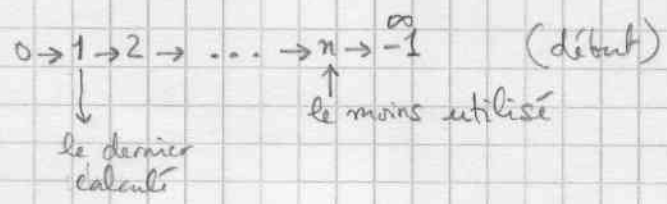
Notes: il peut y avoir des insertions en H, effectuées après la mise de y local

- variables chainées } locales : pointées sur la pile (TVARN ↗)
- index } // : manuel
- littéral local : numérotés à partir de \$FFFF ↓

$n = \text{pile_rem}$

	debut	premier
TMREM	4n	4
4	0	8
8	4...	

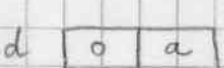
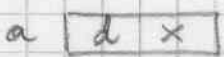
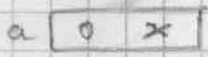
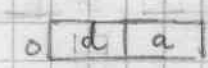
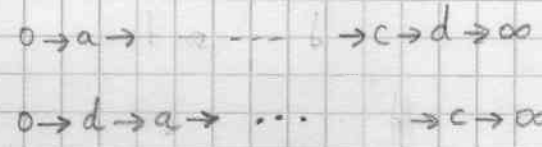
liste liée



TMREM	4(n-1)	0
-------	--------	---



Mémoïsation (au retour de procédure) :



var_d = valeur

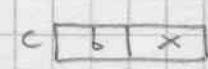
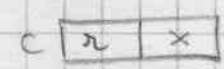
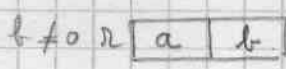
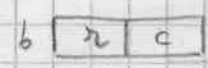
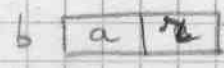
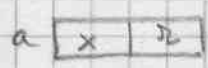


avant

après

(met aussi α en $(\text{TMREM} + d - 1)_L$)

Modification de la liste si r est redne



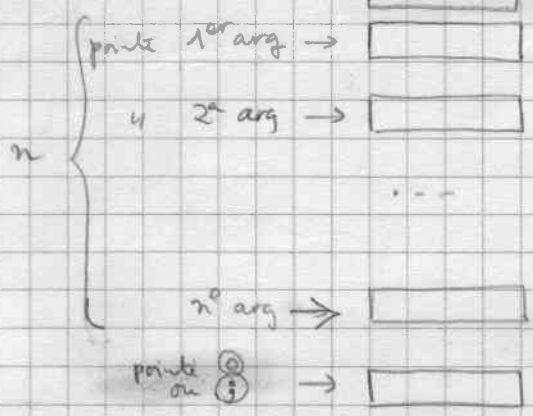
avant

après

avant

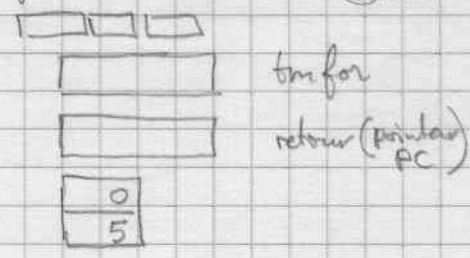
in pile

procédure



local précédent TVARN
 → TVARN / TVARLS
 → TMFOR

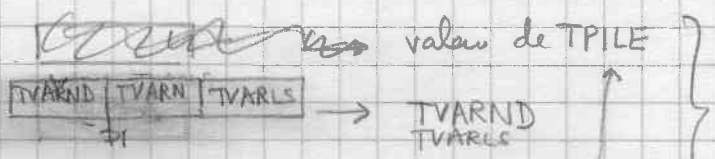
type spec de code 68000
 ③ 121



fonct. { variable
 { chaîne

(in pile)

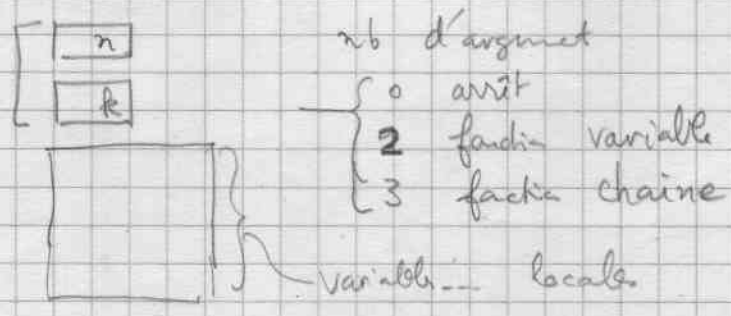
[arrêt 00]



valeurs de
 l'instruction
 appelante



remember { bits = { 0 non normal
 1 option remember



Variable | valeurs

D.B 10, "VALUE"
 @VALU: D.W. \$74, 0
 (type de main)

\$30	si type variable	48
\$40	si type chaîne	64

