

④ [suite de local] de
dataa nomi {, nomi}

```

GED10: BSR   DECAN
        BEQ   MM20  → fin
        TST   D0
        BMI   ERR91  non inconnu

```

X
WNO11 →

```

MOVE.B (A3)+, D0
MEVEN  A3 ← MOVE (A3), -(SP)  type de l'index
MOVE   D0, D1
AND    #F, D1
BEQ    GED14  → pas indicé
CMP.B  #"", (A5)
BEQ    GED14  → avec indicé
MOVE.L A5, -(SP)
LEA    GED12, A5  "(min)"
BSR    WNO11A
MOVE.L (SP)+, A5
BRA    GED15

```

```

GED14: BSR   WNO11A  → type illégal
GED15: BNE   ERRMT
        AND   #F0, D0
        MOVE  (SP)+, D1  size si index
        CMP   #20, D0
        BNE   GED20

```

d0 = type
d2 = size / numéro
A3 = adr / x
d3 = divar num
d5 = nb d éléments
~~d4 = début du tableau d'index~~

cas index

```

CMP   #4, d2
BCS   GED17
TST.B D2
BNE   ERR11N

```

→ base size 1, 2 et 4, pas en début d'octet (x2010101)

```

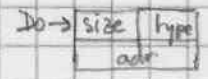
GED17: MOVEM.L D0/A3, -(SP)
        MOVE   D1, (SP)
        MOVE   D1, D0
        MOVE.L D5, D2
        BSR    WNUMIK
        CMP   #4, D2
        BCS   GED18
        TST.B D2
        BNE   GED18
        ADDQ  #1, A3
GED18: MOVEM.L (SP)+, D0/D2
        MOVE.L A3, D5

```

```

GED17: SWAP D0
        MOVE D1, D0
        SWAP D0
        MOVE.L A3, D2
        BRA   GED22

```



```

SUB.L D2, D5
BRA   GED22

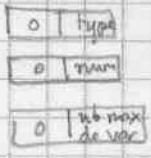
```

cas variable/chaine

```

GEDR0: EXT.L D0
        EXT.L D2
GEDR2: SUB.L D3, D5

```



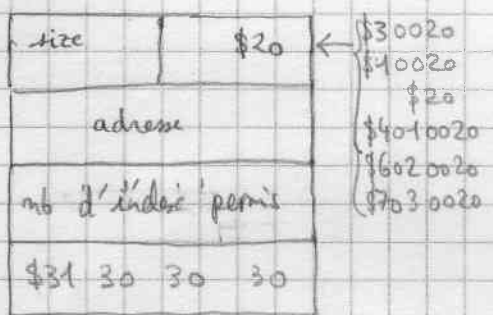
soulève la pile et met les données

```

GEDR2: MOVEM (SP)+, D1/D3/D4
        MOVE.L #31303030, D6      magic
        MOVEM.L D0/D2/D5/D6, -(SP)
        MOVEQ #16, D0
        ADD D0, D3
        ADD D0, D4
        MOVEM D1/D3/D4, -(SP)
        BSR DECCRV
        BEQ GED10
        BRA MM20

```

cas index



cas chaîne/var



local datai \$31303030, 1, ~~\$20~~ ^{\$40A, \$30020} access timer

[suite de local] access nom {, nom}

GED30: BSR DECAN

BEQ MM20 → fin

MOVE 2(SP), D3

→ il manque des données (* local)

SUB #16, D3

X BMT ERRLOC

MOVE D3, 2(SP)

X CMP.L #031303030, 18(SP, D3.W)

BNE ERRLOC → magic (* local)

MOVEM.L 6(SP, D3.W), D3/D4

CMP #20, D3 → type

BNE GED37

index

MOVE.L D3, D4

SWAP D4 size

BSR WINWLOC

cache le nom (si nécessaire)
inscris de la pile_proc
A3 pointe après l'adresse de l'index
à tester (C) d2.L=1
A1 bproc



BNE GED33

BSR MA198

décode les indices d2.L=nb d'indices

GED33: MOVE 2(SP), D3

→ adresse à accéder

MOVEM.L 6(SP, D3.W), D0/D1/D3

CMP.L D3, D2

X BGT ERR1XH

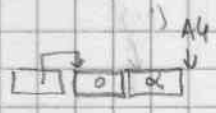
hors du tableau

MOVE.L (A1), A4

MOVE.L -(A4), D3

MOVE.L A4, (A1)

MOVEM.L D1/D3, -(A4)



BTST #0, D1

BEQ GED35

SWAP D0

CMP #1, D0

BEQ ERR0D

CMP #3, D0

BEQ ERR0D

→ adresse impaire

GED35: BSR DECCRV
 BEQ GED30
 BRA MM20

GED37: BSR WNNWLOC

BNE GED38

BSR MA17

GED38: MOVE 2(SP), D3
 CMP 16(SP, D3.W), D2
 BGT ERR1XH
 BRA GED35

variable/chaine D3: type D4: numéro
 ← cache le nom (si nécessaire)
 inscriptie dans pile_proc
 A3 (numéro D4) A2 (type) teste (1) d2.l = 1
 décode (f, l, g, ...) et rajoute la liste d2 = nb d'éléments

hors du tableau