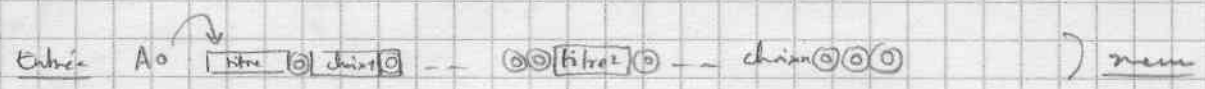
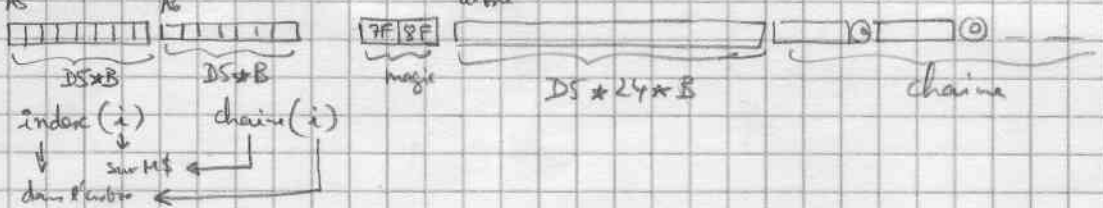



```
XMCONS: MOVEM.L  A0/A5/A6,-(SP)
        BSR    XMKILLA          ôte ancien
        MOVEQ  #4,d0            coordonnés fenetre 0
        MOVE.L d0, ainfin       ainfin+2 → x, y, Δx, Δy
        MOVEQ  #104,d3
        BSR    AESC
        MOVE.L (SP),A0
        MOVEQ  #0,D6            d6= nb de menus
        MOVEQ  #12,D5           d5=nb d'objets      ⊗
        TST.B  (A0)
        BEQ    ERRMEN           → erreur menu
GAG87:  TST.B  (A0)+            ⊗
        BEQ    GAG89
        ADDQ   #1,d5
GAG88:  TST.B  (A0)+
        BNE    GAG88
        BRA    GAG87
GAG89:  ADDQ   #1,d6
        TST.B  (A0)
        BNE    GAG87
        MOVE.L D6,D0            vérif mémoire   d5= $16
        ADD.L  D5,D0                            d6 = 3
        ADD.L  D0,D0
        ADDQ.L #8,d0            marge
        ADD.L  A0,D0
        SUB.L  (SP),D0
        MOVE   D5,D1
        MULU   #24,D1           pour les objets
        ADD.L  D1,D0
        CMP.L  TAMPMENU,D0
        BGT    ERR113           → trop petit
        CMP    #8,d6            ⊗
        BGT    ERRMEN
```




```
        MOVE.L  TMENU,A4
        MOVE.L  A4,A5
        LEA     (A4,DS.L),A6
        MOVE    DS,D0
        SUBQ    #1,D0          ← MOVEQ #-1,d2
GAG91:  MOVE    D2,(A4)+
        DBRA    D0,GAG91
        MOVE    #$7F8F,(A4)+   magic
        MOVE    DS,GARBRN
        MOVE.L  A4,A3
        ADD.L   D1,A3          début de chaine
        MOVE.L  A4,A0
        MOVE.L  A4,-(SP)
        MOVE    DS,D4
        SUBQ    #1,D4
        MOVEQ   #0,d0
        MOVEQ   #$1C,d1
        MOVEQ   #$10,d3
GAG93:  ADDQ    #1,d0          | i de 0 à d5-1 |
        MOVE    D0,(a4)+
        MOVE.L  D2,(a4)+
        MOVE    D1,(a4)+
        CLR.L   (a4)+
        CLR.L   (a4)+
        CLR.L   (a4)+
        MOVE.L  D3,(a4)+
        DBRA    D4,GAG93
        MOVE    #$20,-16(a4)   dernier objet
        MOVE    d2,(a0)+       | i=0 |
        ADDQ    #2,(a0)+
        ADDQ    #3,(a0)+       MOVE #$B,(a0)+
        MOVEM   AINTOUT+2,d0/d1/d2
                               x  y  Δx
        CMP     #2,RESOL
        BCC     GAG95
        ADD     D1,D1
```

{MOVE #$13,GMENHX ⊗ / MOVEQ #$12,D4} Δy

{ADDQ #3,GMENHX ⊗ / MOVEQ #$14,D4}

(voir Bell p38)

structure d'objet i

ob_suit	ob_tête
i+1	-1
ob_taille	ob_type
-1	28 → texte graphique
ob_drap	ob_état
0	0
ob_spec	
0	
ob_x	ob_y
0	0
ob_larg	ob_haut
0	$10

objet 0:

			-1	1
			$B	25 → boîte
x	0	mm 0	0	0
y	$13	B	0	
Δx	$280	$280	0	0
Δy		$80	Δx	400

```
GAG95: SUBQ #3, (a0)+
       ADD  #12, A0
       MOVE d2, (a0)+        Δx
       MOVE $190, (a0)+
       MOVE #$B, (a0)+
       ADDQ #3, (a0)+
       ADDQ #3, (a0)+
       SUBQ #8, (a0)+
       ADDQ #6, A0
       MOVE #$1100, (a0)+
       ADDQ #4, A0
       MOVE D2, (A0)+        Δx
       ADDQ #2, (a0)+   MOVE D4, (A0)+   ⊗
```

$i=1$

$B	2
2	20 → boîte graphique
	$1100
Δx	$12

```
       LEA  9*24(a0), A2
       MOVEQ #$C, d4
       MOVE.L d4, (a2)+
       MOVE.L A2, -(SP)
       SUBQ.L #3, (a2)+
       ADDQ #8, a2
       MOVE d0, (a2)+
       MOVE d1, (a2)+
       MOVE d2, (a2)+
       MOVE #$12C, (a2)+
```

$i=B$

	0	$C
		25 → boîte
α		
	x	y
	Δx	300

α

x

y

Δx

```
       SUBQ #2, (a0)+
       ADDQ #4, (a0)+
       ADDQ #3, (a0)
       ADD  D6, (a0)+
       SUBQ #3, (a0)+
       ADDQ #8, A0
       MOVE D3, (a0)+
       ADDQ #2, A0
       MOVE.L A0, -(SP)
       ADDQ.L #3, (A0)+   { ADDQ #2, A0
                            MOVE GMENHX, (A0)+   ⊗
```

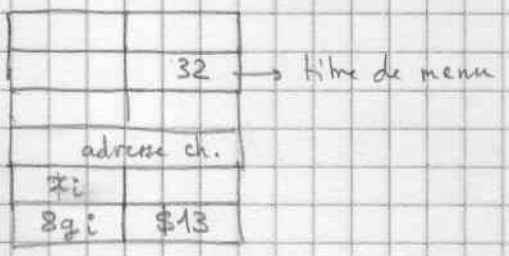
$i=2$

	1	3
	2+d6	25 → boîte
	$10	
β	xi	$13

β

```
        MOVEQ  #0, d6          n° de la chaine M$
        MOVE.L 12(SP),A4       M$
        MOVEQ  #0, d4          x
GAG97:  MOVEQ  #32, d0
        MOVE   (A0)+, d1
        BSR    XMTAM
        ADDQ   #4, A0
        MOVE   D0, (a0)+
        ADDQ   #4, A0
        MOVE.L A3, (a0)
        MOVE.B d0, (a3)+
        MOVEQ  #8, d1
GAG98:  MOVE.B (a4)+, (a3)+
        DBEQ   d1, gag98
        MOVE.B d0, -1(a3)
        MOVE.L A3, d1
        SUB.L  (a0)+, d1       g
        CLR.B  (a3)+
        SUBQ   #1, A4
GAH10:  TST.B  (a4)+
        BNE    GAH10
        MOVE   D4, (A0)+       xi
        MOVE   D4, D3
        ASL    #3, D1          } 8g
        MOVE.L D1, (A0)+       }
        ADD    D1, D4          }
        ADDQ   #3, (A0)+       } ⊗
        MOVE   GMENHX, (A0)+   }
```

boucle sur les menus tà m



```
        MOVE.L  A2,-(SP)
        MOVE    (A2)+,(A2)+
        SUBQ.L  #8,(A2)+
        ADDQ    #4,A2
        MOVE.L  #$FF1100,(A2)+
        ADD     #$10,D3
        MOVE    D3,(A2)+
        ADDQ    #6,A2
        MOVEQ   #0,D2
        MOVEQ   #0,D3
GAH12:  TST.B   (A4)+
        BEQ     GAH20
        ADDQ    #8,A2
        CMP.B   #"_",-(A4)
        BNE     GAH14
        ADDQ    #8,(A2)
GAH14:  ADDQ    #2,A2
        MOVE.L  A3,(A2)
GAH16:  MOVE.B  (A4)+,(A3)+
        BNE     GAH16
        MOVE.L  A3,D0
        SUBQ.L  #1,D0
        SUB.L   (A2)+,D0
        CMP.L   D0,D2
        BCC     GAH18
        MOVE.L  D0,D2
GAH18:  MOVE.L  D3,(A2)+
        ADD     #$10,D3
        ADDQ    #4,A2
        BRA     GAH12
```

$x_i + \$10$

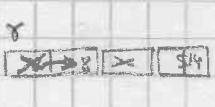
k
h

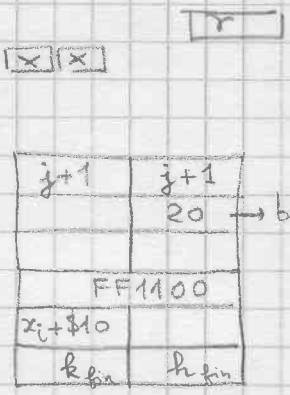
TST.B (A4)
BEQ ERR-MEN "menu"

boucle sur les choix

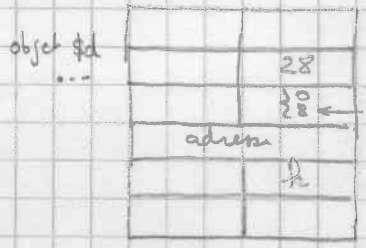
→ fin

MOVE (A2)+, d1
BSR XMTAM





pour le choix d'un menu



d6 : n° de la chaîne
d1 : n° d'index + 1

met la table A5/A6 et avance d6

```
XMTAM:SUBQ #1,d1
      MOVE.B  d1,(A5,d6)
      MOVE.B  d6,(A6,d1.W)
      ADDQ    #1,d6
      RTS
```

```
GAH20: ASL  #3,D2                    k
       ADDQ #1,D6
       MOVE.L (SP),A1                γ

       MOVE -24(A2),D0                   10

       MOVE D0,(A1)+
       MOVE (A1)+,D1                 D

       SUBQ #1,D0                    F
       MOVE D0,(A1)+

       ADD  #4,A1
       MOVE D2,(A1)+                 k
       MOVE D3,(A1)+                 h

GAH22: CMP.L A2,A1
       BGE  GAH24
       ADD  #20,A1
       MOVE D2,(A1)+                 k
       ADDQ #2,A1
       BRA  GAH22

GAH24: SUBQ #1,D1                    C
       MOVE D1,-24(A2)

       MOVE.L (SP)+,A1               γ
       TST.B (A4)
       BNE  GAG97
       MOVE #$B,(A1)                 ⊗
       MOVE.L (SP)+,A1          } β
       MOVE D4,(A1)          x: }
       CMP  -24(A1),D4
       BGE  ERRMEN                   → "menu"

       MOVE.L (SP)+,A1               α
       MOVE D1,(A1)
       MOVE #2,-24(a0)               ⊗
```

```
        MOVEM.L (SP)+,A0/A1/A5/A6          arbre
        MOVE.L  A0, GARBRE
        MOVE.L  A0, D2

        CMP  #2,RESOL                      rectifie hauteurs
        BCC  GAH27                         si base ou moyenne résolution

GAH26:  ADD  #18,A0
        MOVE (A0),D0
        ASR #1,D0
        MOVE D0,(A0)+
        ADDQ #2,A0
        MOVE (A0),D0
        ASR  #1,D0
        MOVE D0,(A0)+
        CMP.L A2,A0
        BLT  GAH26

GAH27:  MOVEQ #1,D1
        BRA  GAG82
```