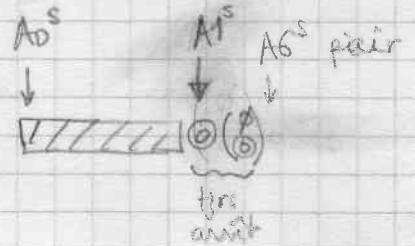
① Pose en libre la chaine « représentant |[A0]|




```
XPRTA1: MOVE  (A0),D0
        CMP   #$4000,D0
        BNE   MH34
⊗ MH33: MOVE.L A6,A0          ← LEA 1(A0),A1
        MOVE  #$3000,(A6)+
        RTS
MH34:   MOVE.L A6,-(SP)
        :AND  #$5FFF,D0
        BCLR  #14,D0
        BEQ   MH38
        MOVEQ #2,D0

MH38:   MOVE  TBASE,D1        ← ADDQ #8,A6
        ADD   D0,D0
        ADD   D0,A6
        CMP.B #16,D1
X       BEQ   MH42            → cas hexa
        BCC   MH39
        ADD   D0,A6
        CMP.B #10,D1
        BEQ   MH51            → cas décimal
        CMP.B #4,D1
        BCC   MH39
        ADD   D0,D0
        ADD   D0,A6

MH39:   BSR   VERA6
        MOVE.L A6,A1
        CLR.B -(A1)
        MOVEM.L A0/A1,-(SP)
        LEA   TBASE,A0
        BSR   XPOSE
        MOVE.L (SP)+,A0
        MOVE.L A2,A1
        BSR   XPOSE
        MOVE.L A2,A0
```

(1)

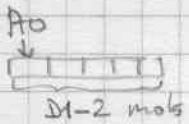
x b

```
MH40: MOVEM.L  A0/A1,-(SP)
      BSR  XDIV1
      MOVE.L (SP)+,A4   x
      MOVE.L (SP)+,A1   b
      MOVE.L (SP)+,A0   c
      MOVE  (A3),D0
      CMP.B #10,D0
      BCS   MH400
      ADD   #7,D0
MH400: ADD  #"0",D0
      MOVE.B D0,-(A0)
      MOVE.L A0,-(SP)
      MOVE.L A4,A0
      BSR  XLB76
      MOVE.L A4,A0
      CMP  #$4000,(A0)
      BNE  MH40
      MOVEM.L (SP)+,A1/A2
      MOVE.L A2,A0 ⟶ libre
      CMP.B #$3A,(A1)
      BCS  MH41
      MOVE.B #"0",-(A1)
MH41 : MOVE.B (A1)+,(A2)+
      BNE  MH41
MH410: LEA  -1(A2),A1
      CLR.B (A2)+
      MOVE.L A2,D0
      BCLR #0,D0
      MOVE.L D0,A6
      RTS
```

① cas hexa

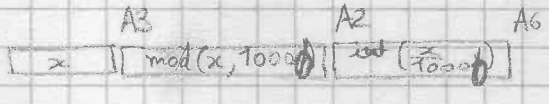
```
MH42: BSR  VERAG   <-- MOVE.L (SP),A6
      MOVEQ #2,D1      MOVE #$30,D4   "0"
      MOVE (A0)+,D0    MOVE #9,D5
      AND #$5FFF,D0    MOVE #$F,D6
      BCLR #14,D0
      BNE  MH43
      MOVE D0,D1
      MOVE (A0)+,D0
MH43 :MOVEQ #3,D2      sortie de D0.W   D0.W 1er mot
                                        D1.W = nb de mots +2 restant
MH44 :ROL #4,D0
      MOVE D0,D3
      AND  D6,D3
      BEQ  MH47
      CMP  D5,D3
      BLE  MH46
      MOVE.B D4,(A6)+   précédé de 0
      ADDQ #7,D3
MH46: ADD  D4,D3
      MOVE.B D3,(A6)+   ⊗
      BRA  MH50
MH47 :DBRA D2,MH44
      BRA  ERRFAT  <-- MH470:MOVE (A0)+,D0
                           MOVEQ #3,D2
répète MH44
MH48 :ROL #4,D0
      MOVE D0,D3
      AND  D6,D3
      CMP  D5,D3
      BLE  MH49
      ADDQ #7,D3
MH49: ADD  D4,D3
      MOVE.B D3,(A6)+   ⊗
MH50: DBRA D2,MH48

   5 :SUBQ #2,D1
      BHI  MH470        ↓ fin
      MOVE.L (SP)+,A0
      MOVE.L A6,A2
      CLR.B (A2)+
      BRA  MH410
```



① cas décimal

```
C10000 : D.W  ~~$43E8~~ ~~2,10000~~ 2,10000   ⊗
MH51: BSR   VERA6
      MOVE.L   A6,A1
      CLR.B  -(A1)
      MOVEM.L  A1/A6,-(SP)
      BSR   XPOSE
MH52: LEA   C10000,A1      10000
      MOVE.L  4(SP),A0     x
      BSR   XDIV1
      MOVE  (A3)+,D0
      BCLR  #14,D0
      BNE   MH53
      MOVE  (A3)+,D0
```



D0, ~~L~~ ∈ [0, ~~9999~~ 9] reste

```
MH53: MOVE  #"0",D1
      MOVE.L  (SP)+,A1   <- { MOVEQ  #~~1~~ 2,D2   ⊗
                              { MH54: EXT.L  D0
      DIVU  #10,D0
      SWAP  D0
      ADD   D1,D0
      MOVE.B  D0,-(A1)
      SWAP  D0
      DBRA  D2,MH54
      ADD   D1,D0
      MOVE.B  D0,-(A1)
      CMP   #$4000,(A2)
      BEQ   MH55          -> fin
      MOVE.L  (SP),A0
      BSR   XLB76
      MOVE.L  A1,-(SP)
      BRA   MH52
MH55: CMP.B  ~~D1,~~(A1)+,D1   enlève les zéros en tête
      BEQ   MH55
      SUBQ  #1,A1
      MOVE.L  (SP)+,~~D0~~/A0   ⊗
      MOVE.L  A0,A2
      BRA   MH41
```