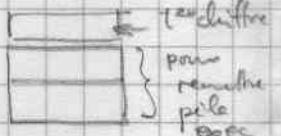
(5) det (f , N[k])

f(i,j) pour i,j ∈ [k,N] ⊂ [0 , 999]
doit renvoyer une expr
(f peut être une variable à 2 indices ou une fonction ou A*g etc : c'est f(i,j) qui doit être une expr valable sans (9))

recopie PB80+4

```
YDET : LEA   TMEXCP, A1          Inscrire f(  11blancs  dans pile exec
       MOVE.L (A1)+, A0  ← MOVEM.L A0/A1,-(SP)
       MOVE.L (A1), A2          } il faut A0 < A2
       SUB    D1, A2  ← MOVE #32,D1   } (ouverge de f...)
       BRA    MG82
MG81 : CMP.L  A2, A0
       BCC    ERRPXC
       MOVE.B D0, (A0)+
MG82 : MOVE.B (A5)+, D0
       BEQ    ERR1SR
       CMP.B  D1, D0
       BEQ    MG82
       CMP.B  #"9", D0
       BNE    MG81
       MOVE.B #"(", (A0)+ ← MOVE.B #"f", (A0)+
       MOVE.L A0, -(SP)
       MOVE   #10, D0
MG83 : MOVE.B D1, (A0)+
       DBRA   D0, MG83
       MOVE.L A5, A2
       BSR    PB880             y termine l'inscription de pile exec
       MOVE.L A2, A5
       BSR    WINDEX            (N) BSR XIPRIOR
       CMP    #1000, D3
       BCC    ERRRG             si N ∉ [0,999]
       MOVE.L D3, D2
       BSR    WVGK              décode , k
       CMP.L  D3, D2
       BCS    ERRRG             si k ∉ [0,N]
       MOVEM.L D2/D3/A5, -(SP)
       LEA    TVARN, A4
   (A) BSR    PUSHNU
       MOVE   (A4), -(SP)
   (B) BSR    PUSHNU
       MOVE   (A4), D1          MOVE TVARN, -(SP)
       MOVE   D1, -(SP)
       ADDQ   #1, D1
M(1,1) MOVE   D1, -(SP)
```



(5)

```
        MOVE.L   6(SP),D0        k
MG84 :  MOVE.L   D0,-(SP)
        MOVE.L   18(SP),A0       1er chiffre
        BSR  PINTA
        MOVE.B  #",",(A0)+
        MOVE.B  #"ξ",(A0)+
        MOVE.L   10(SP),D0       k
MG85 :  MOVEM.L  D0/A0,-(SP)
        BSR   PINTA
        MOVE.B  #")",(A0)+
        MOVE.L   30(SP),A5       début f(i,j)
        BSR   WEXPR              <-- BSR XLPRIOR
        MOVEM.L  (SP)+,D0/A0
        ADDQ.L  #1 D0
        CMP.L    6(SP),D0
                 (N)
        BLE   MG85
        MOVE.L  (SP)+,D0
        ADDQ.L  #1,D0
        CMP.L   2(SP),D0
                (N)
        BLE   MG84
        MOVE  (SP)+,D0           (A)
        MOVEM.L  (SP)+,D1/D2
        SUB.L   D2,D1            D1 = dim - 1

        BSR   XDET
        MOVE.L  (SP)+,A5
        MOVEM.L  (SP)+,D0/A0/A1
        MOVE.L  A0,-(A1)
        RTS
```