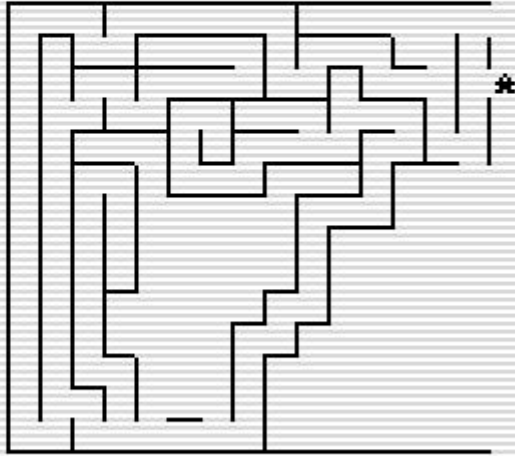


## Wiwo Dido, mapped maze



**This game is based on “Dungeon of Ymir”. It is also an extension of Wiwo Dido's game, my first 1K hires game. After 8 years I wanted the largest maze visible in 1K on screen. This is the result.**

```
; Wiwo Dido's mapped maze
; Game 58 in 1K hires for the ZX81
```

```
? * TORNADO *
```

```
ORG #4009
DUMP 49161
```

```
maxx EQU 15
maxy EQU 14
```

```
size EQU 60
```

```
screen EQU #4240
makemove EQU #401A
```

```
basic LD H,lbuff/256
JR init0
```

```
DEFB 236,212,28
DEFB 126,143,0,18
```

```
eline DEFW last
chadd DEFW last-1
xptr DEFW 0
stkbot DEFW last
```

```

stkend    DEFW last
berg      DEFB 0
mem        DEFW 0
           DEFB 0

init1     JP    init

lastk     DEFB 255,255,255
margin    DEFB 55
nxtlin    DEFW basic
init0     XOR   A
           DEFB #16                ; LD D,

flagx     DEFB 64
strlen    LD    E,A                ; DE now #4000
           LD    B,A
taddr     DEFW 0
seed      LD    L,lbuf*256/256    ; HL now LBUF

frames    DEFW #DD*256+1
coords    LD    HL,hr            ; set IX
sposn     JR    init1

cdflag    DEFB 64

down      INC   B
           CALL fieldtest
           RET   NC
           DEC   B
;          RET                    exit through UP saves a byte

up         CALL NC,fieldtest      ; only C when from down
           RET   C
           DEC   B
           RET

right      INC   C
           CALL flagx
           RET   NC
           DEC   C
;          RET                    same trick with left and right

left       CALL NC,flagx
           RET   C
           DEC   C
           RET

start      LD    A,191
           IN    A,(254)
           RRA
           JR    C,start          ; wait for NewLine

sets       LD    HL,screen+16
           LD    (HL),0           ; clear the visible screen
           INC   HL
           LD    A,H
           CP    #44
           JR    NZ,sets

; clear the maze
           LD    HL,maze          ; clear a previous maze
           LD    B,size
clmaze     LD    (HL),255

```

```

        INC    HL
        DJNZ   clmaze

yloop   LD     B,maxy
xloop   LD     C,maxx
        PUSH  BC
        DEC   B
        DEC   C
        LD    E,1                ; signal no move possible

nxy     LD     A,4
        CALL  rnd                ; 0-3
        OR    D                  ; 4-7
        LD    D,A                ; some start in rnd direction
nf      PUSH  BC
        LD    A,D
        AND   3                  ; make it 0-3
        JR    Z,l1               ; 0 is left
        DEC   A
        JR    Z,u1               ; 1 is up
        DEC   A
        JR    Z,r1               ; 2 is right

dl      INC    B                  ; so down remains
        DEFB  62
u1      DEC    B                  ; up
        DEFB  62
r1      INC    C                  ; right
        DEFB  62
l1      DEC    C                  ; left

usetest LD    A,maxy-1
        CP     B
        JR     C,movedone        ; Y out of size, both ways
        INC    A
        PUSH  DE                  ; save rnd step D and E result
        CP     C                  ; test X out of size both ways
        CCF
        LD    D,4                ; on random field test all dir

dloop   PUSH  DE
        LD    A,D
        CALL  C,makemove          ; when max X no test
        POP   DE
        JR    NC,exitest          ; Also NC when max X, drop DE
        DEC   D
        JR    NZ,dloop

exitest POP   DE                  ; get moved direction
        JR    NC,movedone
        LD    A,D
        XOR    2                  ; opposite direction
        CALL  makemove            ; impossible, get field
setf    LD    A,(HL)              ; set direction
        AND   D
        LD    (HL),A
        LD    E,H
        POP   HL                  ; drop bc
        JR    nxy                ; further from new XY

movedone DEC   D                  ; test next direction
        POP   BC
        JR    NZ,nf

```

```

        POP  BC                ; get original XY
        DEC  E
        JR   NZ,xloop         ; we could move, so try again
        DEC  C                ; goto next field
        JR   NZ,xloop

        DJNZ yloop            ; goto next row
; maze now defined

        CALL rnd-2
        LD   C,D
        LD   B,A
        CALL field
        RES  1,(HL)            ; set exit
        RES  0,(HL)
        CALL rnd-2
        LD   B,A                ; set rnd y
        CALL rnd-2
        LD   C,A                ; set rnd x

gameloop LD   HL,screen-32
        LD   DE,32
        PUSH BC
        INC  B
findy    ADD  HL,DE
        DJNZ findy
        ADD  HL,BC                ; HL now visible field
        POP  BC
        PUSH HL
        CALL up
        POP  HL
        LD   (HL),255            ; show no move up
        JR   C,noup
        INC  B                ; undo move when
        LD   (HL),128           ; there is a way up

noup     LD   DE,16
        ADD  HL,DE

        LD   (wiwopos+1),HL      ; here is where wiwo is

        PUSH HL
        CALL left
        POP  HL
        SET  7,(HL)              ; show no move left
        JR   C,noleft
        INC  C                ; undo left move
        LD   (HL),0             ; show passage left

noleft   INC  HL

        LD   A,C
        CP   maxx                ; test step out right
        JP   Z,start            ; if so, new game

        PUSH HL
        CALL right
        POP  HL
        SET  7,(HL)              ; signal no pass right
        LD   DE,15
        JR   C,noright
        DEC  C                ; undo right move
        LD   (HL),D             ; show pass right
noright  ADD  HL,DE

```

```

LD    A,H
CP    #44
JR    Z,skipdown      ; bottom is done by HR

PUSH  HL
CALL  down
POP   HL
LD    (HL),255        ; show no pass bottom
JR    C,nodown
DEC   B               ; undo move down
LD    (HL),128        ; show pass bottom
nodown INC   HL
SET   7,(HL)          ; show right bottomcorner

skipdown LD   HL,lastk
LD       A,(HL)
wchange CP   (HL)
JR       Z,wchange    ; wait for change in keypress

; controls
LD       A,%11111100  ; combined AZ-read
IN       A,(254)
LD       L,up*256/256
RRA      ; check A
JR       NC,callhl
RRA      ; check Z (or S)
LD       L,down*256/256
JR       NC,callhl
LD       A,%11011111  ; read Y-P
IN       A,(254)
RRA      ; check P
LD       L,right*256/256
JR       NC,callhl
RRA      ; check O
LD       L,left*256/256
callhl CALL NC,#7D    ; a valid move is done
JR       gameloop     ; continue in loop

rnd      LD   A,maxy
LD       HL,(frames)  ; frames
PUSH    DE
rseed    LD   DE,0
ADD     HL,DE
POP     DE
LD      D,A
INC     HL
LD      A,H
AND     #1F
LD      H,A
LD      (rseed+1),HL
LD      A,(HL)
frnd     SUB  D
JR      NC,frnd
ADD     A,D
RET

fieldtest LD  D,191
CALL    field
RLCA
RLCA
RET

field    LD   A,C

```

```

LD    HL,maze-4
LD    A,C
RRCA
RRCA
AND    7
ADD    A,L
LD    L,A
LD    A,B
INC    A
fy     CALL #4014                ; save 2 bytes
JR     NZ,fy
LD    A,C
AND    3
LD    E,A
LD    A,(HL)
fpos   RET    Z
RLCA
RLCA
RRC    D
RRC    D
DEC    E
JR     fpos

hr      LD    B,241                ; fill top with blanc lines
delhr   EX    (SP),HL
EX      (SP),HL
DJNZ   delhr

LD      HL,screen
LD      B,maxy

bloop   EXX
LD      BC,#4009                ; wiwoudg in LBUF
wiwopos LD      HL,0
EXX
LD      C,7
CALL    #4000
LD      DE,16
ADD     HL,DE
RET     C                        ; never true

cloop   PUSH   HL
POP     HL

EXX
LD      A,(BC)
XOR     (HL)
LD      (HL),A
INC     C
EXX
CALL    #4000

RET     C
RET     C

DEC     C
JR      NZ,cloop

ADD     HL,DE
DJNZ    bloop

PUSH    HL
POP     HL
LD      A,I

```

```

        LD    HL,screen

        CALL #4000

        CALL #292
        CALL #220

        LD    IX,hr
        JP    #2A4

; space is the room for the stackpointer
SPACE    EQU  screen+4-size -$
        DEFS space

maze     EQU    $

; 1 time use only, some data and routines stored over sysvar
init     LD    SP,maze
        LD    C,36
        LDIR

        LD    HL,movetab
        LD    C,4
        LD    E,nxtlin*256/256
        LDIR

        LD    HL,fieldtest2
        LD    DE,flagx
        LD    C,7
        LDIR

        LD    HL,screen
        LD    B,15
set2     LD    (HL),255
        INC   HL
        DJNZ  set2
        LD    (HL),128
        INC   HL
set3     LD    (HL),128
        INC   HL
        LD    A,H
        CP    #44
        JR    NZ,set3
        JP    start

movetab  DEFB  left*256/256
        DEFB  up*256/256
        DEFB  right*256/256
        DEFB  down*256/256

lbuf     JP    #C003                ; 4000

lbuf2    LD    A,H                ; 4003
        LD    I,A                ; 4004
        LD    A,L                ; 4006
        LD    R,A                ; 4007

wiwoudg  DEFB  0                  ; 4009
        DEFB  8,28,42,34,42,54    ; 400a

        DEFB  0,0                ; 4010
        DEFW  0                  ; 4012

```

```

save2      INC    HL            ; 4014
           INC    HL
           INC    HL
           INC    HL
           DEC    A
           RET              ; 4019

makemovec  AND     3            ; 401a
           LD     HL,nxtlin    ; 401c
           ADD    A,L          ; 401f
           LD     L,A          ; 4020
           LD     L,(HL)       ; 4021
           JP     (HL)         ; 4022

fieldtest2 LD     D,127
           CALL  field
           RLCA
           RET

last       EQU    $

```