

```
WindowWidth = DisplayWidth
WindowHeight = DisplayHeight
global maxghost , level , WinX , WinY
level = -1
maxghost = 7
WinX = WindowWidth
WinY = WindowHeight
nomainwin
"maze consts & vars
global empty , wall , pil , super , totpil
global maxx , maxy , joystickJN$ , lastkey$
"ghost consts & vars
global hunt , flee , caught , warning , qtel
"player vars
global px , py , pdx , pdy , punten
dim sx( maxghost ) , sy( maxghost )
dim sdx( maxghost ) , sdy( maxghost ) , st( maxghost )
readjoystick 1
if Joy1x 0 and Joy1y 0 then
confirm "Use joystick ?" ; joystickJN$
end if
loadbmp "empty" , DefaultDir$ + "\BMP\empty.bmp"
loadbmp "wall" , DefaultDir$ + "\BMP\wall.bmp"
loadbmp "pil" , DefaultDir$ + "\BMP\pil.bmp"
loadbmp "super" , DefaultDir$ + "\BMP\super.bmp"
loadbmp "warning" , DefaultDir$ + "\BMP\warning.bmp"
loadbmp "hunt" , DefaultDir$ + "\BMP\hunt.bmp"
loadbmp "caught" , DefaultDir$ + "\BMP\caught.bmp"
loadbmp "flee" , DefaultDir$ + "\BMP\flee.bmp"
loadbmp "human" , DefaultDir$ + "\BMP\human.bmp"
hunt = 0 : flee = 1 : caught = 2 : warning = 3
empty = 0 : wall = 1 : pil = 2 : super = 3
dim maze( 20 , 20 )
open "PacMan" for graphics as #m '
#m "trapclose [quit]"
#m "when characterInput [key]"
for x = 0 to 19
for y = 0 to 19
#m "addsprite maze" + nr$( x ) + nr$( y ) + " empty wall pil super"
next y
next x
for i = 0 to maxghost
#m "addsprite ghost" + nr$( i ) + " warning hunt caught flee"
next i
#m "addsprite player human"
call nextlevel
```

```
call drawmaze
" timer 250 , [tmr]
wait
[tmr]
call updatemaze
call drawmaze
wait
[key]
lastkey$ = Inkey$
wait
function sign( x )
if x if x > 0 then sign = 1
sign = 0
end function
function nr$( no )
nr$ = right$( "00" + str$( no ) , 2 )
end function
sub updatemaze
for i = 0 to level
"for all ghosts
select case st( i )
"look at state of ghost
case hunt
if rnd( 0 ) dx = sign( px - sx( i ) )
dy = sign( py - sy( i ) )
if maze( sx( i ) + dx , sy( i ) ) = wall then
dx = 0
end if
if maze( sx( i ) , sy( i ) + dy ) = wall then
dy = 0
end if
if dx = 0 and dy = 0 then
call moveghost i , dx , dy
end if
else
call moveghost i , dx , dy
end if
case flee
if rnd( 0 ) dx = -sign( px - sx( i ) )
dy = -sign( py - sy( i ) )
if maze( sx( i ) + dx , sy( i ) ) = wall then
dx = 0
end if
if maze( sx( i ) , sy( i ) + dy ) = wall then
dy = 0
end if
```

```
if dx = 0 and dy = 0 then
call moveghost i , dx , dy
end if
else
call moveghost i , dx , dy
end if
case else
"case caught & warning
dx = 0
dy = 0
end select
if px = sx( i ) and py = sy( i ) then
"if player spot = ghost spot
if st( i ) = hunt then
notice chr$( 13 ) + "Your Dead !!" _
+ chr$( 13 ) + "Points : " + str$( punten )
close #m
end
end if
if st( i ) = flee then
punten = punten + 20
end if
end if
"update ghost spot
sx( i ) = sx( i ) + dx
sy( i ) = sy( i ) + dy
"this i wil expain in a future version
sdx( i ) = dx
sdy( i ) = dy
"move ghost into maze if its out of it
if sx( i ) if sx( i ) > maxx - 1 then sx( i ) = 2
if sy( i ) if sy( i ) > maxy - 1 then sy( i ) = 2
next i
if joystickJN$ = "yes" then
readjoystick 1
dx = 0
if Joy1x > 256 * 256 * 2 / 3 then
dx = 1
end if
if Joy1x dx = -1
end if
dy = 0
if Joy1y > 256 * 256 * 2 / 3 then
dy = 1
end if
if Joy1y dy = -1
```

```
end if
else
dx = 0
if lastkey$ = "6" then
dx = 1
end if
if lastkey$ = "4" then
dx = -1
end if
dy = 0
if lastkey$ = "2" then
dy = 1
end if
if lastkey$ = "8" then
dy = -1
end if
lastkey$ = ""
end if
if maze( px + dx , py + dy ) = wall then
dx = 0
dy = 0
end if
if maze( px , py ) = pil then
punten = punten + 1
totpil = totpil - 1
maze( px , py ) = empty
end if
if maze( px , py ) = super then
qtel = 400
totpil = totpil - 1
maze( px , py ) = empty
end if
"update player spot
px = px + dx
py = py + dy
"this i wil explain in future vesions
pdx = dx
pdy = dy
"move player into maze if he is out of it
if px if px > maxx - 1 then px = 2
if py if py > maxy - 1 then py = 2
"update flee time
if qtel > 0 then
qtel = qtel - 1
if qtel for i = 0 to level
st( i ) = warning
```

```
next i
end if
end if
if totpil maxx then
maxx = len( m$ )
end if
for x = 1 to len( m$ )
mi$ = mid$( m$ , x , 1 )
select case mi$
case "m"
maze( x , maxy ) = wall
case "."
maze( x , maxy ) = pil
totpil = totpil + 1
case "@"
maze( x , maxy ) = super
totpil = totpil + 1
case else
maze( x , maxy ) = empty
end select
" print maze( x , y )
'Checks the maze() array is filling correctly. It is, now, with 0,1,2 or 3 in all locs.
next x
print
wend
for i =0 to level
sx( i ) = 9
sy( i ) = 7
st( i ) = hunt
next i
px = 9
py = 11
[dat]
" 12345678901234567
data " m m "
data " mmmmmmmm mmmmmmmm "
data " m@...m m...@m "
data " m.mmm.m m.mmm.m "
data " m.m.....m.m "
data " m.m.mmm.mmm.m.m "
data " m...m....m...m "
data " mmmm.m.mmm.m.mmmm "
data " ..... "
data " mmmm.m.mmm.m.mmmm "
data " m...m....m...m "
data " m.m.mmm.mmm.m.m "
```

```
data " m.m.....m.m "
data " m.mmm.m m.mmm.m "
data " m@...m m...@m "
data " mmmmmmmm mmmmmmmm "
data " m m "
data "="
end sub
sub drawmaze
for x = 1 to 17 ' NB there are only 17x17!!! See the mainwin printout..
for y = 1 to 17
whatismaze = maze( x , y )
'Shows maze() containing all '0' except for two '1' - which is WRONG! Why???
select case whatismaze
case wall
#m "spriteimage maze" + nr$( x ) + nr$( y ) + " wall"
case pil
#m "spriteimage maze" + nr$( x ) + nr$( y ) + " pil"
case super
#m "spriteimage maze" + nr$( x ) + nr$( y ) + " super"
case else
#m "spriteimage maze" + nr$( x ) + nr$( y ) + " empty"
'For testing only- it shows array not being read or filled correctly.
" #m "backcolor red"
" #m "place "; x *tx; " "; y *ty
" #m "ellipsefilled 10 6"
end select
#m "spritexy maze" + nr$( x ) + nr$( y ) ; " " _
; ( x - px ) * 64 + WinX / 2 - 64 ; " " _
; ( y - py ) * 64 + WinY / 2 - 64
next y
print
next x
for i = 0 to level
whatisghost = st( i )
select case whatisghost
case hunt
#m "spriteimage ghost" + nr$( i ) + " hunt"
case flee
#m "spriteimage ghost" + nr$( i ) + " flee"
case caught
#m "spriteimage ghost" + nr$( i ) + " caught"
case else "warning"
#m "spriteimage ghost" + nr$( i ) + " warning"
end select
#m "spritexy ghost" + nr$( i ) ; " " _
; ( sx( i ) - px ) * 64 + WinX / 2 - 64; " " _
```

```
; ( sy( i ) - py ) * 64 + WinY / 2 - 64
next i
#m "spritexy player " ; WinX / 2 - 64; " " ; WinY / 2 - 64
#m "drawsprites"
end sub

[quit]
close #m
end
```