一、 學習成果(程式功能說明)

1. macro.h

```
PrintStr macro string ;輸出字串
        mov ah,09h
        mov dx,offset string
        int 21h
        endm

GetChar macro           ;等待輸入
        mov ah,10h
        int 16h
        endm

SetMode macro mode      ;設定顯示模式
        mov ah,00h
        mov al,mode
        int 10h
        endm

SetColor macro color    ;設定背景色
        mov ah,0bh
        mov bh,00h
        mov bl,color
        int 10h
        endm

WrPixel macro col,row,color    ;寫入像點
        mov ah,0ch
        mov bh,00h
        mov al,color
        mov cx,col
        mov dx,row
        int 10h
        endm
```

```
SetCursor macro row,col    ;設定游標位置
        mov dh,row
        mov dl,col
        mov bx,00h
        mov ah,02h
        int 10h
        endm


MUS_RESET macro        ;滑鼠重置
        mov ax,0000h
        int 33h
        endm


MUS_HIND macro         ;隱藏滑鼠游標
        mov ax,0002h
        int 33h
        endm

SET_MUSmacro  Col,Row   ;設定滑鼠游標位置
        mov ax,0004h
        mov dx,Row
        mov cx,Col
        int 33h
        endm

MUS_range_x macro max,min    ;設定滑鼠水平游標的範圍
        mov ax,0007h
        mov dx,max
        mov cx,min
        int 33h
        endm
```

```
MUS_range_y macro max,min    ;設定滑鼠垂直游標的範圍
        mov ax,0008h
        mov dx,max
        mov cx,min
        int 33h
        endm
```

2. team6.asm

```
include macro.h

.model large
.data
Start_word db 'Press ENTER to start or ESC to exit',10,13,'$'
Score_H dw 0
thing_row dw 0
thing_col dw 0
thing_size dw 30
thing_col_right dw 0
thing_col_left dw 0
thing_row_top dw 0
thing_row_down dw 0
bullet_row dw 0
bullet_col dw 0
bullet_size dw 10
bullet_col_right dw 0
bullet_col_left dw 0
bullet_row_top dw 0
bullet_row_down dw 0
count dw 0
color db 0
lose dw 0
speed dw 1
End_word db 'Press ENTER to restart or ESC to exit',10,13,'Your
Score:$'
```

```
.stack
.code
pnt_thing proc              ;印方塊
  intit_set:
     mov cx,thing_col
     mov dx,thing_row
     mov count,0
     mov di,thing_size
  Print:
     WrPixel cx,dx,color
     dec di
     inc cx
     cmp di,0
     ja Print
  next_row:
     inc count
     mov di,count
     cmp di,thing_size
     ja over
     mov cx,thing_col
     mov di,thing_size
     inc dx
     jmp Print
  over:
     mov cx,thing_col
     mov dx,thing_row
     ret
pnt_thing   endp

pnt_bullet proc             ;印掉落物
  intit_set:
     mov cx,bullet_col
     mov dx,bullet_row
     mov count,0
```

```
    mov di,bullet_size
  Print:
    WrPixel cx,dx,color
    dec di
    inc cx
    cmp di,0
    ja Print
  next_row:
    inc count
    mov di,count
    cmp di,bullet_size
    ja over
    mov cx,bullet_col
    mov di,bullet_size
    inc dx
    jmp Print
  over:
    mov cx,bullet_col
    mov dx,bullet_row
    ret
pnt_bullet      endp

Delay proc
    mov   cx,1
  L1:
    push cx
    mov cx,65535
  L2:
    loop L2
    pop cx
    loop L1
    ret
Delay  endp

valueToASCII proc
```

```
    mov cx,0
    mov bl,10
  Hex2Asc:
    div bl
    mov dl,ah
    add dl,30h
    push dx
    inc cx
    mov ah,0
    cmp al,0
    jne Hex2Asc
  addSpace:
    cmp cx,3
    je keepPnt
    mov dl,' '
    push dx
    inc cx
    jmp addSpace
  keepPnt:
    pop ax
    PrintChar al
    loop keepPnt
    ret
valueToASCII endp

main proc
    mov   ax,@data
    mov   ds,ax

    PrintStr Start_word   ;寫開始文字
L1: GetChar               ;讀鍵盤
    cmp al,0Dh            ;判斷有沒有按下ENTER
    je L2
    cmp al,1bh     ;判斷有沒有按下ESC
```

```
    je quit
    jmp L1

L2: mov lose,0
    mov Score_H,0
    mov speed,1
    SetMode 12h       ;遊戲開始
    SetColor 00h
    MUS_RESET
    MUS_range_x 609,0  ;設置x邊界範圍
    MUS_range_y 429,0  ;設置y邊界範圍
    SET_MUS 300,300        ;設置起始位置
    MUS_HIND       ;隱藏游標
    jmp tran0

tran0:  SetCursor 0,0      ;寫分數表
    mov ax,Score_H
    call valueToASCII
    SetCursor 1,0
    mov ax,lose        ;寫未接住次數
    call valueToASCII
    SetCursor 8,0
    mov ax,speed       ;寫掉落速度
    call valueToASCII

bullet0:    cmp lose,5     ;判斷掉落是否有5次
        je quit            ;若5次就結束遊戲
        mov bullet_row,0 ;初始掉落物
        in ax,40h          ;隨機給16bit
        mov dx,0
        mov bx,600         ;範圍限制在0~600
        div bx
        mov bullet_col,dx
```

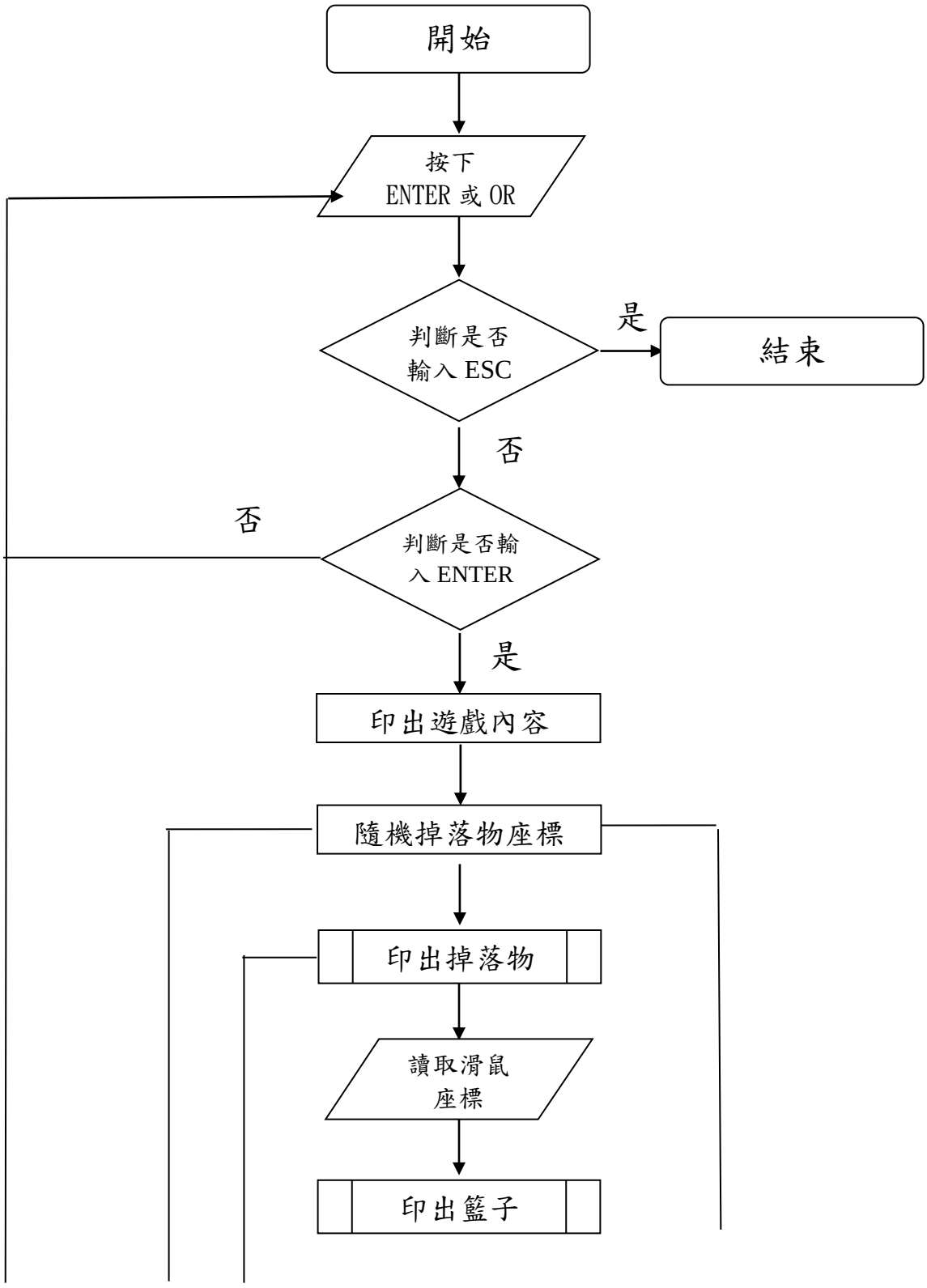
```
        mov bullet_col_left,dx          ;記錄掉落物左邊界
        mov bullet_col_right,dx
        add bullet_col_right,10         ;記錄掉落物右邊界
        mov bullet_row_top,0      ;記錄掉落物上邊界
        mov bullet_row_down,10      ;記錄掉落物下邊界
bullet_start:     mov color,0h
        call pnt_bullet      ;清除原掉落物顏色
        mov cx,speed       ;速度利用加y座標控制
        add bullet_row,cx
        add bullet_row_top,cx;記錄掉落物上邊界
        add bullet_row_down,cx  ;記錄掉落物下邊界
        mov color,0Eh
        call pnt_bullet      ;開始畫掉落物

L3: MUS_GET03             ;定位滑鼠
    mov thing_col,cx       ;儲存方塊左上座標
    mov thing_row,dx
    mov color,0Eh
    call pnt_thing             ;畫方塊
    call Delay
    mov color,0h
    call pnt_thing             ;清除原方塊顏色
    mov thing_col_left,cx      ;記錄方塊左邊界
    mov thing_col_right,cx
    add thing_col_right,30          ;記錄方塊右邊界
    mov thing_row_top,dx            ;記錄方塊上邊界
    mov thing_row_down,dx
    add thing_row_down,30           ;記錄方塊下邊界

    SetCursor 0,0       ;寫分數表
    mov ax,Score_H
    call valueToASCII
```

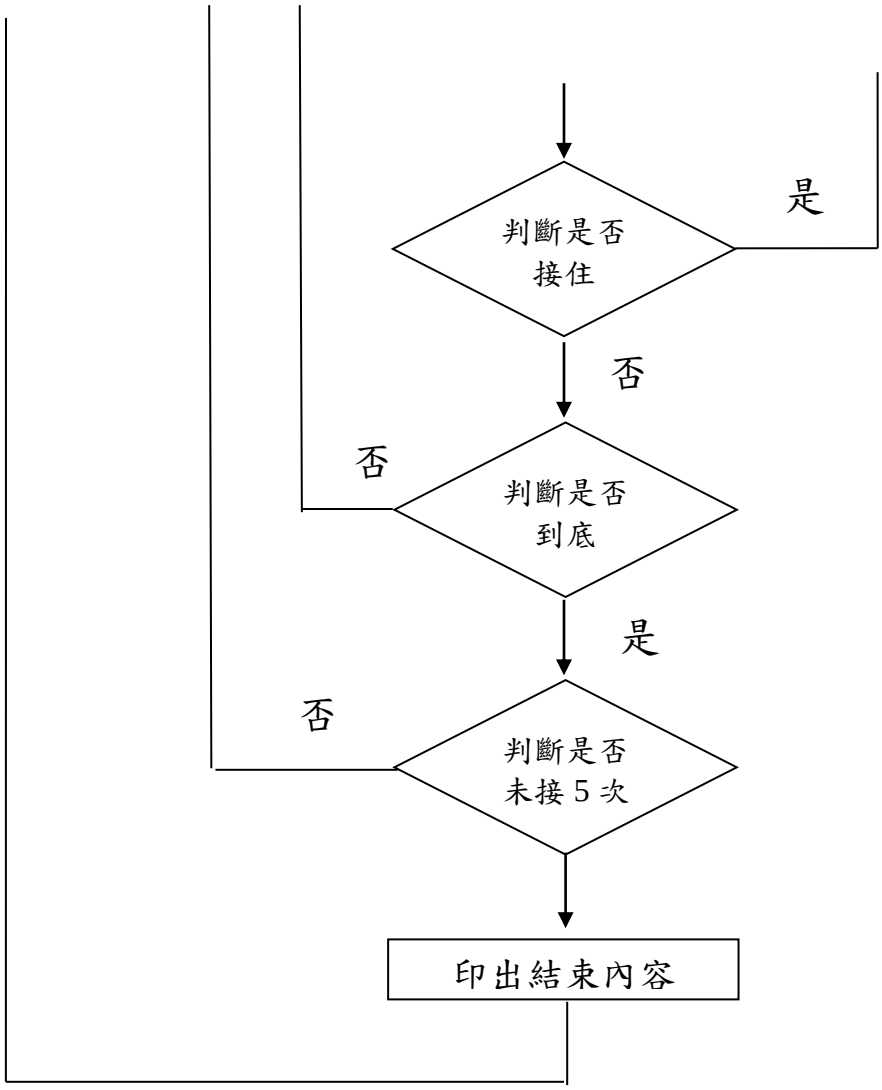
```
SetCursor 1,0
mov ax,lose         ;寫未接住次數
call valueToASCII
SetCursor 2,0           ;寫方塊左邊界
mov ax,thing_col_left
call valueToASCII
SetCursor 3,0           ;寫方塊右邊界
mov ax,thing_col_right
call valueToASCII
SetCursor 4,0           ;寫方塊上邊界
mov ax,thing_row_top
call valueToASCII
SetCursor 5,0           ;寫方塊下邊界
mov ax,thing_row_down
call valueToASCII
call Delay
SetCursor 6,0           ;寫掉落物x座標
mov ax,bullet_col_left
call valueToASCII
SetCursor 7,0           ;寫掉落物y座標
mov ax,bullet_col_right
call valueToASCII
SetCursor 8,0           ;寫掉落物x座標
mov ax,bullet_row_top
call valueToASCII
SetCursor 9,0           ;寫掉落物y座標
mov ax,bullet_row_down
call valueToASCII
SetCursor 10,0              ;寫掉落物速度
mov ax,speed
call valueToASCII

mov ah,06h          ;判斷是否按ESC提前結束
```

```
    mov dl,0ffh
    int 21h
    cmp al,1bh
    je quit

    mov ax,thing_col_right        ;判斷是否在右邊界右邊
    cmp ax,bullet_col_right
    jae L7              ;如果是 接下去判斷
    jmp L8              ;如果不是 跳判斷是否到底
L7: mov ax,thing_col_left     ;判斷是否在左邊界左邊
    cmp ax,bullet_col_left
    jbe L9              ;如果是 接下去判斷
    jmp L8              ;如果不是 跳判斷是否到底
L9: mov ax,thing_row_top          ;判斷是否在上邊界下面
    cmp ax,bullet_row_top
    jbe L10                 ;如果是 接下去判斷
    jmp L8              ;如果不是 跳判斷是否到底
L10:    mov ax,thing_row_down       ;判斷是否在下邊界上面
    cmp ax,bullet_row_down
    jae tran          ;如果是 跳加分
L8: cmp bullet_row,470        ;預設底部為470
    jb bullet_start           ;如果還沒到底就跳回繼續畫
    jmp L11                   ;如果到底 跳結束此次畫掉落物
tran:   inc Score_H               ;加分
    inc speed           ;加掉落物速度
    SetCursor 10,0              ;寫掉落物速度
    mov ax,speed
    call valueToASCII
    SetCursor 0,0           ;寫分數
    mov ax,Score_H
    call valueToASCII
    mov color,0h
```

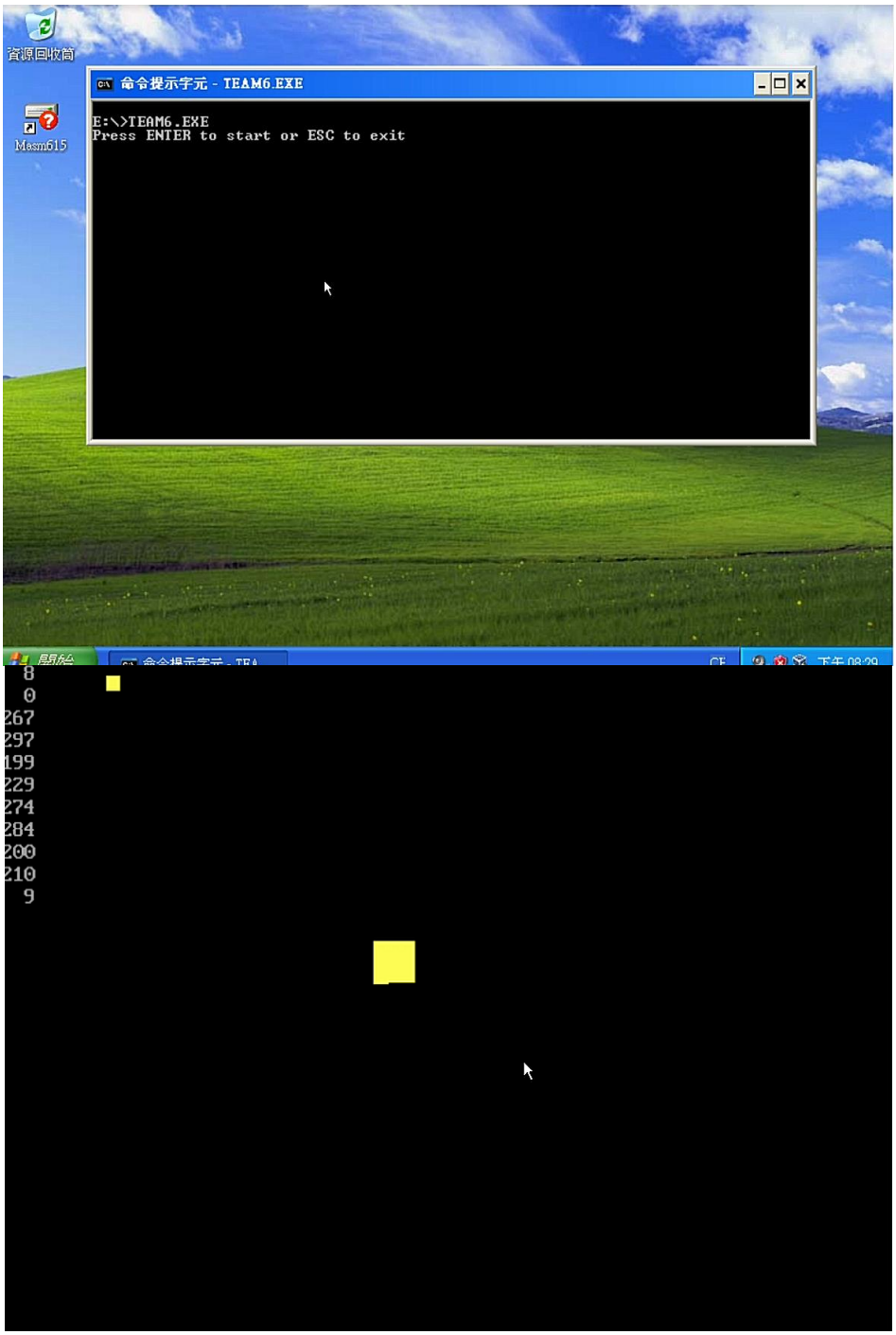
```
    call pnt_bullet             ;清除原掉落物顏色
    jmp bullet0                 ;跳新的掉落物
L11:  mov color,0h
    call pnt_bullet             ;清除原掉落物顏色
    inc lose                  ;加未接住次數
    SetCursor 1,0               ;寫未接住次數
    mov ax,lose
    call valueToASCII
    jmp bullet0                 ;跳新的掉落物


quit:   SetMode 03h         ;結束部分 跳回文字模式
    PrintStr End_word       ;寫結束文字
    mov ax,Score_H          ;寫最後得分
    call valueToASCII
quit1:  GetChar             ;判斷是否要再玩一次
    cmp al,0Dh
    je L2
    cmp al,1bh
    je quit2
    jmp quit1
quit2:  mov   ax,4c00h
        int  21h
main endp
end main
```

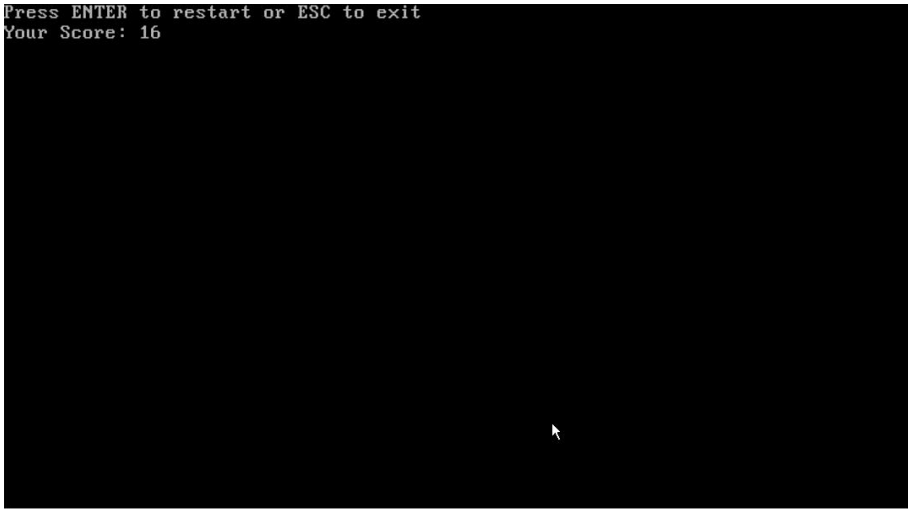
二、 流程圖


開始
按下
ENTER 或 OR
判斷是否
輸入 ESC
是
結束
否
否
判斷是否輸
入 ENTER
是
印出遊戲內容
隨機掉落物座標
印出掉落物
讀取滑鼠
座標
印出籃子


判斷是否
接住
是
否
判斷是否
到底
否
是
判斷是否
未接 5 次
否
印出結束內容

三、實習結果


資源回收筒
Masm615
命令提示字元 - IEAM6.EXE
E:\>IEAM6.EXE
Press ENTER to start or ESC to exit
8
0
267
297
199
229
274
284
200
210
9


Press ENTER to restart or ESC to exit
Your Score: 16