```
[org 0x0100]
jmp start
; subroutine to clear the screen
clrscr:
         push es
         push ax
         push di
         mov ax, 0xb800
         mov es, ax ; point es to video base
         mov di, 0 ; point di to top left column
nextloc:
        mov word [es:di], 0x0720 ; clear next char on screen
        add di, 2 ; move to next screen location
        cmp di, 4000 ; has the whole screen cleared
        jne nextloc ; if no clear next position
        pop di
        pop ax
        pop es
        ret
; subroutine to print a string at top left of screen
; takes address of string and its length as parameters
print_circle:
        push bp
        mov bp, sp
        push es
        push ax
        push cx
        push si
        push di

        mov ax,[bp+4]
        mov bl,80
        mul bx
        add ax,[bp+6]
        mov di, ax ; point di to top left column
        mov ax, 0xb800
        mov es, ax ; point es to video base
        mov cx,1
        mov ah, 0x07 ; normal attribute fixed in al
        mov al,'*'
nextchar:
        mov [es:di], ax ; show this char on screen
        loop nextchar ; repeat the operation cx times
        mov ax,[bp+4]
        add ax,1
        mov bl,80
        mul bx
        mov cx,[bp+6]
        sub cx,1
        add ax,cx
        mov di, ax ; point di to top left column
        mov ax, 0xb800
        mov es, ax ; point es to video base
        mov cx,1
        mov ah, 0x07 ; normal attribute fixed in al
        mov al,'*'
        mov [es:di], ax ; show this char on screen
        pop di
        pop si
        pop cx
        pop ax
        pop es
        pop bp
        ret 4
start:
         call clrscr ; call the clrscr subroutine
        mov ax, 70
        push ax ; x_axis
        mov ax,6        ;y_axis
        push ax
        call print_circle
        mov ax, 0x4c00 ; terminate program
        int 0x21
```