

4.

a) Read an alphanumeric character and display its equivalent ASCII code at the center of the

screen.

```
.model small
```

```
.data
```

```
msg db"Enter the key",10,13,"$"
```

```
.code
```

```
mov ax,@data
```

```
mov ds,ax
```

```
mov ah,00h
```

```
mov al,03h
```

```
int 10h
```

```
mov dx,offset msg
```

```
mov ah,09h
```

```
int 21h
```

```
mov ah,01h
```

```
int 21h
```

```
mov bl,al
```

```
mov ah,02h
```

```
mov bh,00h
```

```
mov dh,12
```

```
mov dl,40
```

```
int 10h
```

```
mov al,bl
```

```
and al,0f0h
```

```
mov cl,04h
```

```
shr al,cl
```

```

call display
mov al,bl
and al,0fh
call display
mov ah,4ch
int 21h
display PROC
cmp al,0ah
jb skip
add al,7h
skip: add al,30h
mov dl,al
mov ah,02h
int 21h
RET
display ENDP
end

```

b) Display messages FIRE and HELP alternately with flickering effects on a 7-segment display

interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the

messages (Examiner does not specify these delay values nor is it necessary for the student to

compute these values).

```
.model small
```

```
.stack
```

```
.data
```

```
.code
```

```
again : lea si,fire
fire db 86H,88H,0f9H,8eH
help db 8cH,0c7H,86H,89H
PA equ 9800H
PB equ 9801H
PC equ 9802H
CR equ 9803H
mov ax,@data
mov ds,ax
mov al,80H
mov dx,CR
out dx,al
mov bh,0ah
call display
call delay
lea di,help
call display
call delay
dec bh
cmp bh,00H
je exit
jmp again
mov ah,4cH
int 21H
display PROC
mov cx,04H
loop1 : mov bl,08H
```

```
mov al,[si]
next : rol al,01H
mov dx,PB
out dx,al
push ax
mov al,0FFH
inc dx
out dx,al
mov al,00H
out dx,al
dec bl
pop ax
jnz next
jmp next1
next1: inc si
loop loop1
RET
display ENDP
delay PROC
push cx
push bx
mov cx,0FFFFH
up : mov bx,0FFFFH
up1 : dec bx
jnz up1
loop up
pop bx
```

```
pop cx
```

```
RET
```

```
delay ENDP
```

```
end
```