# TASK 1:

org 0x0100

start:

mov ah, 00h ; keyboard services

int 16h ; Wait for a key press

cmp ah, 01h ; Check if the Escape key is pressed

je exit

cmp ah, 3Ah ; Check if a character key is pressed

jbe check\_key

jmp start ; Wait for another key press

check\_key:

mov dl, al ; Store the ASCII value of the key in dl

cmp dl, 'h' ; Check if the key is 'h'

je display\_key

cmp dl, 'u' ; Check if the key is 'u'

je display\_key

cmp dl, 'z' ; Check if the key is 'z'

je display\_key

cmp dl, 'a' ; Check if the key is 'a'

je display\_key

cmp dl, 'i' ; Check if the key is 'i'

je display\_key

cmp dl, 'f' ; Check if the key is 'f'

je display\_key

; Key is not one of the valid letters

mov dx, invalid\_msg

mov ah, 09h ; Select function 09h of int 21h (display string)

int 21h ; Display the invalid\_msg

jmp start ; Wait for another key press

display\_key:

mov dx, key\_msg

mov ah, 09h ; Select function 09h of int 21h (display string)

int 21h ; Display the key\_msg

mov dl, al ; Store the ASCII value of the key in dl

mov ah, 02h ; Select function 02h of int 21h (display character)

int 21h ; Display the character

jmp \_start ; Wait for another key press

exit:

mov dx, esc\_msg

mov ah, 09h ; Select function 09h of int 21h (display string)

int 21h ; Display the esc\_msg

mov ah, 4Ch ; Select function 4Ch of int 21h (exit program)

int 21h ; Terminate the program

section .data

key\_msg db "Key pressed:", 0

invalid\_msg db "Invalid key.", 0

esc\_msg db "Esc key pressed. Exiting.", 0

# TASK 2:

int 0x0100

start:

; Display the prompt to enter the first number

mov ah, 9

mov dx, num1\_msg

int 21h

; Get the first number from the user

mov ah, 0x01

int 21h

; Check if the first number is within the range 5

cmp al, 5

jl invalid\_num

cmp al, 6

jg invalid\_num

; Store the first number in a variable

mov bl, al

; Display the prompt to enter the second number

mov ah, 9

mov dx, num2\_msg

int 21h

; Get the second number from the user

mov ah, 0x01

int 21h

; Check if the second number is within the range 5

cmp al, 5

jl invalid\_num

cmp al, 6

jg invalid\_num

; Store the second number in a variable

mov cl, al

; Calculate the sum of the two numbers

add bl, cl

mov al, bl

; Display the sum of the two numbers

mov ah, 9

mov dx, sum\_msg

int 21h

; Display the sum value

mov ah, 0x02

mov dl, al

int 21h

; Exit the program

mov ah, 0x4c

int 0x21

num1\_msg db "Enter the first number (5-6): "

num2\_msg db "Enter the second number (5-6): "

sum\_msg db "The sum of the two numbers is: "

invalid\_num db "Invalid number. Please enter a number within the range 5-6."