**AREEBA NADEEM**

**57027**

**Task # 1**

1. Write an assembly language program that find the square of a number entered by user. Print the result on screen.

.model small

.data

'

.code

main proc

mov ah,1

int 21h

sub al,48

mov bl,al

mov al,bl

mul bl

AAM

mov ch,ah

mov cl,al

mov dl,ch

add dx,48

mov ah,2

int 21h

mov dl,cl

add dx,48

mov ah,2

int 21h

mov ah,4ch

int 21h

main endp

end main

**Output**

****

1. Write an assembly language program that find the cube of a number entered by user. Print the result on screen.

**.model small**

**.data**

**.code**

**main proc**

**mov ah,1**

**int 21h**

**sub al,48**

**mov bl,al**

**mov bl,al**

**mul bl**

**mul bl**

**AAM**

**mov ch,ah**

**mov cl,al**

**mov dl,ch**

**add dx,48**

**mov ah,2**

**int 21h**

**mov dl,cl**

**add dx,48**

**mov ah,2**

**int 21h**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**

****

**Task number 3:**

**Perform multiplication of two numbers entered by user. Print the result on screen.**

**; You may customize this and other start-up templates;**

**; The location of this template is c:\emu8086\inc\0\_com\_template.txt**

**org 100h**

**.model small**

**.data**

**.code**

**main proc**

**mov ah,1**

**int 21h**

**sub al,48**

**mov bl,al**

**mov ah, 01**

**int 21h**

**sub al, 48**

**mul bl**

**AAM**

**mov ch,ah**

**mov cl,al**

**mov dl,ch**

**add dx,48**

**mov ah,2**

**int 21h**

**mov dl,cl**

**add dx,48**

**mov ah,2**

**int 21h**

**mov ah,4ch**

**int 21h**

**main endp**

**end main**

**ret**

**output**

****

**Task 4:**

1. Write an assembly language program that finds the area of a triangle.

**.model small**

**.stack 100h**

**.data**

**msg db 'entre baes= $'**

**msg1 db 'entre height= $'**

**msg2 db 'result= $'**

**.code**

**main proc**

**mov ax,@data**

**mov ds,ax**

**mov dl,msg**

**mov dx,offset msg**

**mov ah,9**

**int 21h**

**mov ah,1**

**int 21h**

**mov bl,al**

**sub bl,48**

**mov dl,10**

**mov ah,2**

**int 21h**

**mov ax,@data**

**mov ds,ax**

**mov dl,msg1**

**mov dx,offset msg1**

**mov ah,9**

**int 21h**

**mov ah,1**

**int 21h**

**mov cl,al**

**sub cl,48**

**mov dl,10**

**mov ah,2**

**int 21h**

**mov dl,13**

**mov ah,2**

**int 21h**

**mov dl,13**

**mov ah,2**

**int 21h**

**mov ax,@data**

**mov ds,ax**

**mov dl,msg2**

**mov dx,offset msg2**

**mov ah,9**

**int 21h**

**mov al,bl**

**mov bl,2**

**mul cl**

**div bl**

**mov dl,al**

**add dl,48**

**AAM**

**mov ch,ah**

**mov cl,al**

**mov dl,ch**

**add dl,48**

**mov ah,2**

**int 21h**

**mov dl,cl**

**add dl,48**

**mov ah,2**

**int 21h**

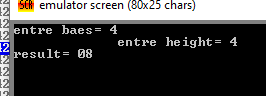
**mov ah, 4ch**

**int 21h**

**main endp**

**end main**

**Output:**

****