

## Assignment #1

## Computer Organization and Assembly Language

### Question 1:

We want to print the pattern like the one below on screen (a space filled diamond) :

\*

\* \*

\* \*

\* \* For size N=7=number of rows

\* \*

\* \*

\*

1. Write a code segment that can input number of rows of pattern from the user and store it in a variable named *N* .
2. Supposing that N is odd , print the pattern of required size on screen

(You can treat both of the above as independents)

**SOLUTION:**

org 100h

.data

c1 db 0

counter db 1

ms db -1

check db 0

half db 0

control db 0

move db 0

.code

mov ah,1

int 21h

sub al,30h

mov cl,al

mov bl,al

mov move,al

shr cl,1

mov half,cl

add half,2

mov c1,cl

mov ch,cl

add ch,3

mov bh,-1

top:

mov control,0

mov ah,2

mov dl,0x0A

int 21h

mov ah,2

mov dl,0x0D

int 21h

mov al,half

cmp counter,al

jl down

inc ch

mov c1,ch

jmp space

down:

dec ch

mov c1,ch

space:

mov ah,2

mov dl,' '

int 21h

dec c1

jnz space

jz stars

stars:

mov ah,2

mov dl,'\*'

int 21h

cmp control,1

je end

midspace:

cmp counter,1

je end

cmp counter,bl

je end

mov ah,2

mov dl,' '

int 21h

dec ms

jnz midspace

inc control

jmp stars

end:

mov al,half

inc counter

cmp counter,al

jl end1

sub bh,2

jmp end2

end1:

add bh,2

end2:

mov ms,bh

dec move

jnz top

ret

### Question 2:

Write a code segment that can :

1. Continuously inputs characters from the user until the user presses carriage return . Your program should also count the number of characters inputted . (Assume we input them into an array whose name is STRING)
2. (A separate code segment) Goes through the above the STRING and then tells how many each of the vowels are there . Store the number of A’s in variable *A\_Count* , number of E’s in *E\_Count* and so on .
3. (A separate code segment) tell how many of the characters are not alphabets.

**SOLUTION:**

org 100h

.data

msg db 'Enter value:$'

vowels db 'a','e','i','o','u'

check db 0

sum db 0

inputs db 0

;string db 6 dup (0)

.code

main proc

call input

call vowelscount

ret

main endp

input proc

mov dx, offset msg

mov ah,9

int 21h

mov si,300h

top:

mov ah,2

mov dl,0x0A

int 21h

mov ah,2

mov dl,0x0D

int 21h

mov ah,1

int 21h

cmp al,0x0D

je end

;sub al,30h

mov [si],al

inc inputs

inc si

jmp top

end:

ret

input endp

vowelscount proc

mov si,300h

scheck:

lea di,vowels

mov cx,5

topv:

mov al,[si]

mov bl,[di]

cmp bl,al

jne ignore

inc sum

ignore:

inc di

loop topv

inc si

dec inputs

jnz scheck

ret