Problem : 01

Write an Assembly Language program to display all alphabet characters

(A-Z) and (a-z).

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

lowerCase db 97

upperCase db 65

.code

printAllCharecters macro case

mov ah,02h

mov cx,26

mov dl,case

@loop:

int 21h

inc dl

dec cx

jnz @loop

endm

main proc

mov ax,@data

mov ds,ax

printAllCharecters lowerCase

mov ah,04ch

int 21h

main endp

end main

Output:

Alphabet A to Z:

ABCDEFGHIJKLMNOPQRSTUVWXYZ

Abcdefghijklmnopqrstuvwxyz

Problem : 02

Write an Assembly Language program to display a Massage N times in

different line.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

massage db 100 dup('$')

n db ?

.code

main proc

mov ax,@data

mov ds,ax

mov si,offset massage

@input:

mov ah,01h

int 21h

cmp al,13

je @endOfInput

mov [si],al

inc si

jmp @input

@endOfInput:

printn ''

mov ah,01h

int 21h

mov n,al

sub n,30h

mov cx,0h

mov cl,n

@for:

printn ''

mov ah,09h

lea dx,massage

int 21h

loop @for

mov ah,04ch

int 21h

main endp

end main

Input:

My name is Saiful Islam Rasel.

10

Output:

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

My name is Saiful Islam Rasel.

Problem : 03

Write an Assembly Language program to read a string and display the length of the string.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

massage db 100 dup('$')

len db ?

.code

main proc

mov ax,@data

mov ds,ax

mov si,offset massage

mov cx,0h

@input:

mov ah,01h

int 21h

cmp al,13

je @endOfInput

inc cl

mov [si],al

inc si

jmp @input

@endOfInput:

mov len,cl

printn ''

mov ah,02h

mov dl,cl

add dl,30h

int 21h

mov ah,04ch

int 21h

main endp

end main

Input:

Rasel

Output:

5

Problem : 04

Write an Assembly Language program to read a string and display the reverse of the string.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

massage db 100 dup('$')

.code

main proc

mov ax,@data

mov ds,ax

mov si,offset massage

mov cx,0h

@input:

mov ah,01h

int 21h

cmp al,13

je @endOfInput

mov [si],al

inc si

inc cl

push ax

jmp @input

@endOfInput:

mov ah,02h

@reverseDisplay:

pop dx

int 21h

loop @reverseDisplay

mov ah,04ch

int 21h

main endp

end main

Input:

Rasel

Output:

Lesar

Problem : 05

Write an Assembly Language program to convert the case of letter.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

lower db 97

upper db 65

.code

main proc

mov ax,@data

mov ds,ax

mov ah,01h

int 21h

mov dl,al

sub dl,32

mov ah,02h

int 21h

printn ''

mov ah,01h

int 21h

mov dl,al

add dl,32

mov ah,02h

int 21h

mov ah,04ch

int 21h

main endp

end main

Input:

a

B

Output:

A

B

Problem : 06

Write an Assembly Language program to read two string and display

concatenation of .two string.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

massage db 100 dup('$')

.code

main proc

mov ax,@data

mov ds,ax

mov si,offset massage

mov cx,0h

@input1:

mov ah,01h

int 21h

cmp al,13

je @endOfInput1

inc cl

mov [si],al

inc si

jmp @input1

@endOfInput1:

printn ''

@input2:

mov ah,01h

int 21h

cmp al,13

je @endOfInput2

inc cl

mov [si],al

inc si

jmp @input2

@endOfInput2:

printn ''

mov ah,09h

lea dx,massage

int 21h

mov ah,04ch

int 21h

main endp

end main

Input:

Saiful Islam

Rasel.

Output:

Saiful Islam Rasel.

Problem : 07

Write an Assembly Language program to input N and check it is even or

odd number.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

n db ?

.code

main proc

mov ax,@data

mov ds,ax

mov ah,01h

int 21h

mov n,al

test n,01h

je @even

printn ''

@odd:

printn 'Odd'

jmp @exit

@even:

printn 'Even'

@exit:

mov ah,04ch

int 21h

main endp

end main

Input:

5

6

Output:

Odd

Even

Problem : 08

Write an Assembly Language program to input N and print factorial of N.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

n dw ?

.code

main proc

mov ax,@data

mov ds,ax

mov ah,01h

int 21h

mov ah,0h

sub al,48

mov n,ax

mov cx,n

mov ax,1h

@fact:

mul cx

loop @fact

printn ''

mov dx,ax

add dx,48

mov ah,02h

int 21h

mov ah,04ch

int 21h

main endp

end main

Input:

3

Output:

6

Problem : 09

Write an Assembly Language program to input and display the sum of a

series of number.

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

n db ?

sum db 0

.code

main proc

mov ax,@data

mov ds,ax

mov ah,01h

int 21h

sub al,48

mov n,al

mov ch,0h

mov cl,n

@inputSeries:

int 21h

sub al,48

add sum,al

loop @inputSeries

mov ah,02h

mov dl,sum

add dl,48

int 21h

mov ah,04ch

int 21h

main endp

end main

Input:

3

1 2 3

Output:

6

Problem : 10

Write an Assembly Language program find the result of the expression

(M+N-P+1).

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

m db 10

n db 5

p db 11

result db ?

.code

main proc

mov ax,@data

mov ds,ax

mov al,m

add al,n

sub al,p

inc al

mov result,al

mov ah,02h

mov dl,result

add dl,48

int 21h

mov ah,04ch

int 21h

main endp

end main

Output:

5

Problem : 11

Write an Assembly Language program to find result of series (1+2+..+N).

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

n db ?

sum db 0

.code

main proc

mov ax,@data

mov ds,ax

mov ah,01h

int 21h

sub al,48

mov n,al

mov dl,n

inc dl

mov ah,0h

mul dl

mov bl,2

div bl

mov sum,al

mov ah,02h

mov dl,sum

add dl,48

int 21h

mov ah,04ch

int 21h

main endp

end main

Input:

3

Output:

6

Problem : 12

Write an Assembly Language program to compute () ).

Solution:

include "emu8086.inc"

.model small

.stack 100h

.data

x db 1,1,2,1

y db 2,1,1,3

n db 4

result dw 0

.code

main proc

mov ax,@data

mov ds,ax

mov ch,0h

mov cl,n

@looping:

mov si,cx

dec si

mov al,x[si]

mul y[si]

add result,ax

loop @looping

mov ah,02h

mov dx,result

add dx,48

int 21h

mov ah,04ch

int 21h

main endp

end main

Output:

8

Problem : 13

Write an Assembly Language program to find the smallest element.

Solution:

INCLUDE "emu8086.inc"

.model small

.stack 100h

.data

array DB 5 dup('?')

small DB ?

.code

main PROC

MOV AX, @data

MOV DS, AX

print "Enter 5 Numbers: "

MOV SI, OFFSET array

MOV CX, 5

@input:

MOV AH, 1

INT 21h

MOV [SI], AL

print " "

INC SI

loop @input

printn " "

MOV SI, OFFSET array

MOV CX, 4

MOV BL, [SI]

MOV small, BL

INC SI

@smallCheck:

MOV BL, [SI]

CMP small, BL

JAE @SWAP

@CONTINUE:

INC SI

loop @smallCheck

JMP @ENDTASK

@SWAP:

XCHG small, BL

JMP @CONTINUE

@ENDTASK:

print "Smallest element: "

MOV AH, 2

MOV DL, small

INT 21h

MOV AH, 4ch

INT 21h

main ENDP

END main

Input:

1 2 3 4 5

Output:

1

Problem : 14

Write an Assembly Language program to sort array element.

Solution:

INCLUDE "emu8086.inc"

.model small

.stack 100h

.data

array DB 5 dup('?')

.code

main PROC

MOV AX, @data

MOV DS, AX

print "Enter 5 Numbers: "

MOV SI, OFFSET array

MOV CX, 5

LOOP1:

MOV AH, 1

INT 21h

MOV [SI], AL

print " "

INC SI

LOOP loop1

printn ""

MOV BX, 5

LEA SI, array

CALL BUBBLE\_SORT

MOV SI, OFFSET array

MOV CX, 5

LOOP2:

MOV AH, 2

MOV DL, [SI]

INT 21h

print " "

INC SI

LOOP lOOP2

printn ""

MOV AH, 4ch

INT 21h

main ENDP

BUBBLE\_SORT PROC

MOV AX, SI

MOV CX, BX

DEC CX

@OUTER\_LOOP:

MOV BX, CX

MOV SI, AX

MOV DI, AX

INC DI

@INNER\_LOOP:

MOV DL, [SI]

CMP [DI], DL

JNG @SKIP\_EXCHANGE

XCHG DL, [DI]

MOV [SI], DL

@SKIP\_EXCHANGE:

INC SI

INC DI

DEC BX

JNZ @INNER\_LOOP

LOOP @OUTER\_LOOP

RET

BUBBLE\_SORT ENDP

END main

Input:

1 3 5 4 2

Output:

1 2 3 4 5