**Coal Programs**

**;PROGRAM TO PRIINT SINGLE CHARACTER**

DOSSEG

.MODEL SMALL

.STACK 100H

.DATA

.CODE

MAIN PROC

MOV DL, 'A'

MOV AH,2

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

**;PROGRAM TO SUBTRACT 2 NUMBERS**

dosseg

.model small

.stack 100h

.data

.code

main proc

mov bl,3

mov cl,1

sub bl,cl

add bl,48

mov dl,bl

mov ah,2

int 21h

mov ah,4ch

int 21h

main endp

end main

**;PROGRAM TO ADD 2 NUMBERS**

dosseg

.model small

.stack 100h

.data

.code

main proc

mov bl, 1

mov cl, 2

add bl,cl

add bl,48

mov dl,bl

mov ah,2

int 21h

mov ah,4ch

int 21h

main endp

end main

**;PROGRAM TO INCREMENT AND PRINT 0 TO 9**

dosseg

.model small

.stack 100h

.data

.code

main proc

mov cx,10

mov dx,48

L1:

mov ah,2

int 21h

inc dx

loop L1

mov ah, 4ch

int 21h

main endp

end main

**;PROGRAM TO ADD 2 NUMBERS INPUT BY A USER**

dosseg

.model small

.stack 100h

.data

.code

main proc

mov ah,1

int 21h

mov bl,al

mov ah,1

int 21h

add bl,al

sub bl,48

mov dl,bl

mov ah,2

int 21h

mov ah,4ch

int 21h

main endp

end main

**;PROGRAM TO PRINT STRINGS ON TWO DIFFERENT LINES**

dosseg

.model small

.stack 100h

.data

msg1 db 'hello$'

msg2 db 'world$'

.code

main proc

mov ax,@data

mov ds,ax

mov dx,offset msg1

mov ah,9

int 21h

mov dx,10

mov ah,2

int 21h

mov dx,13

mov ah,2

int 21h

mov dx,offset msg2

mov ah,9

int 21h

mov ah,4ch

int 21h

main endp

end main

**;Program to print string on new line**

dosseg

.model small

.stack 100h

.data

n\_line db 0ah,0dh, "$" ;for new line

.code

main proc

mov ax, @data

mov ds,ax

mov dl,'a'

mov ah,2

int 21h

lea dx,n\_line

mov ah,9

int 21h

mov dl, 'b'

mov ah,2

int 21h

mov ah,4ch

int 21h

main endp

end main

**;PROGRAM TO PRINT INPUT NUMBER IS EQUAL OR NOT TO GIVEN NUMBER IN PROGRAM**

dosseg

.model small

.stack 100h

.data

msg1 db 'number is equal$'

msg2 db 'number is not equal$'

.code

main proc

mov ax,@data

mov ds,ax

mov dl,'3'

mov ah,1

int 21h

cmp al,dl

je l1

mov dx, offset msg2

mov ah,9

int 21h

mov ah,4ch

int 21h

l1:

mov dx, offset msg1

mov ah,9

int 21h

mov ah,4ch

int 21h

main endp

end main

**;PROGRAM TO CONVERT CAPITAL LETTER INTO SMALL LETTER**

dosseg

.model small

.stack 100h

.data

.code

main proc

mov ah,1

int 21h

mov dl,al

add dl,32

mov ah,2

int 21h

mov ah,4ch

int 21h

main endp

end main

**;program to take input from user and display it**

dosseg

.model small

.stack 100h

.data ;data section

msg1 db 'Please Enter First Number (0-9)$' ;prompt for first number

msg2 db 'Please Enter Second Number (0-9)$' ;prompt for second number

msg3 db 'The sum is$' ;prompt for sum

.code ;code section

main proc

mov ax, @data ;obtain segment access of data

mov ds, ax ;initialize ds

mov dx, offset msg1 ;obtain offset address of string1

mov ah, 9 ;service no. to display a string

int 21h ;call DOS to do operation

mov dx, 10 ;cursor goes to next line

mov ah, 2 ;display

int 21h ;call DOS to to operation

mov dx, 13 ;cursor goes to start of next line

mov ah, 2 ;display

int 21h ;call DOS to do operation

mov ah, 1 ;input string1 by user

int 21h ;call DOS to do operation

mov bl,al ;mov al into bl as it will differntiate b/w two inputs

mov dx, 10 ;cursor goes to next line

mov ah, 2 ;display

int 21h ;call DOS to do operation

mov dx, 13 ;cursor goes to start of next line

mov ah, 2 ;display

int 21h ;call DOS to do operation

mov dx, offset msg2 ;obtain offset address of string2

mov ah, 9 ;service no. to display offset

int 21h ;call DOS to do operation

mov dx, 10

mov ah, 2

int 21h

mov dx, 13

mov ah, 2

int 21h

mov ah, 1 ;display

int 21h ;call DOS to do operatiion

add bl, al ;add values enter by user

sub bl, 48 ;subtract 48 through sum to obtain ASCII value

mov dx, 10

mov ah, 2

int 21h

mov dx, 13

mov ah, 2

int 21h

mov dx, offset msg3 ;obtain offset address of string3

mov ah, 9 ;service no. to display offset

int 21h ;call DOS to do operation

mov dx, 10

mov ah, 2

int 21h

mov dx, 13

mov ah, 2

int 21h

mov dl, bl ;mov bl to dl to show the ouput

mov ah, 2

int 21h

mov ah, 4ch

int 21h

main endp

end main

**;PROGRAM TO PRINT ALL CAPITAL LETTERS FROM A TO Z**

dosseg

.model small

.stack 100h

.data

.code

main proc

mov cx,26

mov dx,65

L1:

mov ah,2

int 21h

inc dx

loop L1

mov ah,4ch

int 21h

main endp

end main

**;Program to divide 2 numbers**

dosseg

.model small

.stack 100h

.data

q db ?

r db ?

.code

main proc

mov ax, 26

mov bl,5

div bl

mov q,al

mov r,ah

mov dl,q

add dl,48

mov ah,2

int 21h

mov dl,r

add dl,48

mov ah,2

int 21h

mov ah,4ch

int 21h

main endp

end main

**;Program to multiply 2 numbers**

dosseg

.model small

.stack 100h

.data

.code

main proc

mov al,9

mov bl,9

mul bl

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

**;Program to print an array using a loop**

dosseg

.model small

.stack 100h

.data

arr1 db 'a','b','c'

.code

main proc

mov ax,@data

mov ds,ax

mov si,offset arr1

mov cx,3

l1:

mov dx,[si]

mov ah,2

int 21h

inc si

loop l1

main endp

end main

**;Program to print length of a string input by user**

dosseg

.model small

.stack 100h

.data

arr db 100 dup('$')

str1 db'Enter the string: $'

str2 db 'The length of the string is: $'

.code

main proc

mov ax,@data

mov ds,ax

mov dx,offset str1

mov ah,9

int 21h

mov bl,0 ;variable for count character in a string

mov si, offset arr ;si point to the array first inl1:

l1:

mov ah,1 ;input string through user

int 21h

cmp al,13 ;untill user press enter key

je stringend

mov [si],al ;store character in an array

inc si ;increment index value in an array

inc bl ;increment count variable as well

jmp l1

stringend:

mov dx,offset str2

mov ah,9

int 21h

mov dl,bl

mov ah,2

add dl,48

int 21h

mov ah,4ch

int 21h

main endp

end main

**;Program that display two digit number stored in bx**

.model small

.stack 100h

.data

string1 db 'The two digit number stored in BX is : $'

.code

main:

mov ax, @data

mov ds, ax

mov dx, offset string1

mov ah, 9

int 21h

mov bx, 69

mov ax, bx

mov bl, 10

div bl

mov bl, ah

mov dl, al

add dl, 48

mov ah, 2

int 21h

mov dl, bl

add dl, 48

mov ah, 2

int 21h

over:mov ah, 4ch

int 21h

end main

**;Program that takes string as input and count its characters and check the given string is odd or even**

.model small ;Both code and data not over 64k

.stack 100h ;256 bytes reserved for stack

.data

string1 db 'Please Type the string here : $' ;20 bytes of string

string2 db 100 dup('$') ;an array inialized with $

string3 db 'The length of the string is : $' ;20 bytes of string

string4 db 'The string is odd $' ;20 bytes of string

string5 db 'The string is even $' ;20 bytes of string

.code

main:

mov ax, @data ;Obtain segment address of data

mov ds, ax ;Initalize DS

lea dx, string1 ;Obtain offset address of string

mov ah, 9 ;Service routine to display a string

int 21h ;Call DOS to do the operation

mov dx, 10 ;Service routine to move the control to next line

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

mov dx, 13 ;Service routine to move the control to start of the line

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

mov cl,0 ;inialized the counter (CL) register

lea si, string2 ;Obtain offset address of string

top:

mov ah, 1 ;Service routine to take input from user

int 21h ;Call DOS to do the operation

cmp al, 13 ;copmare input value with 13(ASCII of ENTER key)

je stringend ;Jump if AL(input) is equal to 13 to stringend label

mov [si],al ;Place the input value in SI(Source Index)

inc si ;Increment in SI to move the array index

inc cl ;Icrement in CL(counter register)

jmp top ;Jump to top label

stringend:

lea dx, string3 ;Obtain offset address of string

mov ah, 9 ;Service routine to display a string

int 21h ;Call DOS to do the operation

mov al, cl ;Place the CL in Al for division

mov ah, 0 ;Make AX register(AH=0,AL=n1+n2)

mov bl, 10 ;Initialize BL with 10

div bl ;Division pf AX by BL (AX/BL)

mov bl, ah ;Avoid overriding in ah in Seervice routine

mov dl, al ;Place the AL in DL to display it

add dl, 48 ;ADD the ASCII value

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

mov dl, bl ;Place the BL in DL to display it

add dl, 48 ;ADD the ASCII value

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

mov ah, 0 ;Make AX register(AH=0,AL=n1)

mov bl, 2 ;Initialize BL with 2

div bl ;Division pf AX by BL (AX/BL)

cmp ah, 1 ;Compare remainder with 1

je nodd ;Jump if remiander is equal to 1 to nodd label

mov dx, 10 ;Service routine to move the control to next line

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

mov dx, 13 ;Service routine to move the control to start of the line

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

lea dx, string5 ;Obtain offset address of string

mov ah, 9 ;Service routine to display a string

int 21h ;Call DOS to do the operation

jmp over ;jump to over label

nodd:mov dx, 10 ;Service routine to move the control to next line

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

mov dx, 13 ;Service routine to move the control to start of the line

mov ah, 2 ;Service routine to display a charcater

int 21h ;Call DOS to do the operation

lea dx, string4 ;Obtain offset address of string

mov ah, 9 ;Service routine to display a string

int 21h ;Call DOS to do the operation

over:mov ah, 4ch ;Service routine to terminate the program

int 21h ;Call DOS to do the operation

end main