**את"מ מעבדה 6:  
  
מגישים:  
 אורי מלכא- 314862996   
אלן ציפין- 313206062**;

; lab6 ori and alan

;

.MODEL SMALL

.STACK 100h

.DATA

.386

;Vars

Stack\_Size DW (?)

Max DW (?)

Min DW (?)

NumMenu DW (?)

NumInput DW 0

Char DB 0

Flag DB 0

;Strings

Menu DB 'Enter 1.push 2.pop 3.min 4.max 5.exit',13,10,'$'

ErrorMenu DB 13,10,'For the Menu you must enter a number between 1-5!!' ,13,10,'$'

ErrorInput DB 13,10,'Its not a number!! please try agian' ,13,10, '$'

StackIsFull DB 13,10,'The stack is full you cannot push!',13,10, '$'

StackIsEmpty DB 13,10,'Error! the stack is empty !!',13,10, '$'

.CODE

ProgStart:

MOV AX,@DATA ; Set DS to point ...

MOV DS,AX ; ... to data segment

MOV Stack\_Size,SP

;Here we want to get a number from 1-5

MenuLoop:

MOV AH,9 ; Set print option for int 21h

MOV DX,OFFSET Menu ; Set DS:DX to point to DisplayString

INT 21h

;Get Char

XOR AX,AX

MOV AH,1

INT 21h

MOV Char,AL

;unitiliaze the loop

XOR BX,BX

MOV BL,'1'

MOV CX,5

CMenu:

CMP Char,BL

JE EndMenu

INC BL

LOOP CMenu

;PRINT ERROR

MOV AH,9 ; Set print option for int 21h

MOV DX,OFFSET ErrorMenu ; Set DS:DX to point to DisplayString

INT 21h ; Print DisplayString

JMP MenuLoop

EndMenu:

XOR AH,AH

MOV NumMenu,AX ;Store the Menu selection

;Check Option '5'

CMP AX,'5'

JE FreeTheStack

;~~~print 13,10~~~~~

MOV AH,2

MOV DL,13

INT 21h

MOV AH,2

MOV DL,10

INT 21h

;~~~print 13,10~~~~~

;CMPS

MOV AX,NumMenu

CMP AX,'1'

JE Option1

CMP AX,'2'

JE Option2

CMP AX,'3'

JE Option3

CMP AX,'4'

JE Option4

;START ~~~~~~~~PUSH #1~~~~~~~~~

Option1:

;Check that the stack is not full

CMP SP,0

JE Stack\_Full

;Input Function Number

InputLoop:

XOR AX,AX

MOV AH,1

INT 21h

MOV Char,AL

;unitiliaze the loop

XOR BX,BX

MOV BL,'0'

MOV CX,10

CInput:

CMP Char,BL

JE EndInput

INC BL

LOOP CInput

;PRINT ERROR

MOV AH,9 ; Set print option for int 21h

MOV DX,Offset ErrorInput ; Set DS:DX to point to DisplayString

INT 21h ; Print DisplayString

JMP InputLoop

EndInput: ;One we got here we have a correct number input , and a menu selection 1-4

;~~~print 13,10~~~~~

MOV AH,2

MOV DL,13

INT 21h

MOV AH,2

MOV DL,10

INT 21h

;~~~print 13,10~~~~~

XOR AX,AX

MOV AL,Char

MOV NumInput,AX ;Store the Input number

;This for the first push to make sure that we are initlazing the max and min

CMP flag,1

JE Proceed

MOV AX,NumInput

MOV Min,AX

MOV Max,AX

INC Flag

Proceed:

;check max

XOR AX,AX

MOV AX,Max

CMP NumInput,AX ;NumInput?Max

JBE NotMax

MOV AX,NumInput

MOV Max,AX

NotMax:

;check min

XOR AX,AX

MOV AX,Min

CMP NumInput,AX ;NumInput?Min

JAE NotMin

MOV AX,NumInput

MOV Min,AX

NotMin:

;Once we got here our current maximum is Max, current minimum is Min and NumInput=AX

;Our stack sturcture is built like this:

;max

;min

;num

PUSH NumInput

PUSH Min

PUSH Max

JMP MenuLoop

Stack\_Full:

MOV AH,9 ; Set print option for int 21h

MOV DX,OFFSET StackIsFull ; Set DS:DX to point to DisplayString

INT 21h

JMP MenuLoop

;END ~~~~~~~~PUSH #1~~~~~~~~~

;START ~~~~~~~~POP #2~~~~~~~~~

Option2:

;We check that the stack is not empty

MOV AX,Stack\_Size

CMP AX,SP

JE Stack\_Empty

POP AX ;we free AX=Max so far

POP AX ;we free AX=Min so far

POP DX ;popped number

MOV AH,2 ;DL is already has the popped number

INT 21h

MOV AH,2

MOV DL,13

INT 21h

MOV AH,2

MOV DL,10

INT 21h

JMP MenuLoop

Stack\_Empty:

MOV AH,9 ; Set print option for int 21h

MOV DX,OFFSET StackIsEmpty ; Set DS:DX to point to DisplayString

INT 21h

MOV Flag,0 ;this line is to make sure that user cannot empty the stack by popping out and then push with out initliazing min and max

JMP MenuLoop

;END ~~~~~~~~POP #2~~~~~~~~~

;START ~~~~~~~~MIN #3~~~~~~~~~

Option3:

;We check that the stack is not empty

MOV AX,Stack\_Size

CMP AX,SP

JE Stack\_Empty\_Min

;We print the minimum

POP AX

MOV Max,AX

POP AX

MOV Min,AX

MOV AH,2

MOV DX,Min

INT 21h

MOV AH,2

MOV DL,13

INT 21h

MOV AH,2

MOV DL,10

INT 21h

PUSH Min

PUSH Max

JMP MenuLoop

Stack\_Empty\_Min:

MOV AH,9 ; Set print option for int 21h

MOV DX,OFFSET StackIsEmpty ; Set DS:DX to point to DisplayString

INT 21h

MOV Flag,0 ;this line is to make sure that user cannot empty the stack by popping out and then push with out initliazing min and max

JMP MenuLoop

;END ~~~~~~~~MIN #3~~~~~~~~~

;START ~~~~~~~~MAX #4~~~~~~~~~

Option4:

;We check that the stack is not empty

MOV AX,Stack\_Size

CMP AX,SP

JE Stack\_Empty\_Max

;We print the maximum

POP Max

MOV AH,2 ;DL is already has the popped number

MOV DX,Max

INT 21h

MOV AH,2

MOV DL,13

INT 21h

MOV AH,2

MOV DL,10

INT 21h

PUSH Max

JMP MenuLoop

Stack\_Empty\_Max:

MOV AH,9 ; Set print option for int 21h

MOV DX,OFFSET StackIsEmpty ; Set DS:DX to point to DisplayString

INT 21h

MOV Flag,0 ;this line is to make sure that user cannot empty the stack by popping out and then push with out initliazing min and max

JMP MenuLoop

;END ~~~~~~~~MAX #4~~~~~~~~~

FreeTheStack:

MOV AX,Stack\_Size

LoopToFree:

CMP AX,SP ;ax==sp when stack is empty.

JE Finish

POP BX

JMP LoopToFree

Finish:

MOV AH,4Ch ; Set terminate option for int 21h

INT 21h ; Return to DOS (terminate program)

END ProgStart

