Coal Assignment # 3

Ahmed Kasteer

20F-0336

Section: 3D

TASK 1:

a)

INCLUDE Irvine32.inc

.data

X dd ?

string1 db "ebx: ",0

string2 db "ecx: ",0

string3 db "X: ",0

string4 db "if (ebx > ecx) then X=1",0

.code

main PROC

lea edx, string4

call writeString

call crlf

lea edx, string1

call writeString

call readDec

mov ebx,eax

lea edx, string2

call writeString

call readDec

mov ecx,eax

cmp ebx,ecx

jg greater

jmp done

greater:

mov X, 1

mov eax,X

lea edx, string3

call writeString

call writeDec

jmp done

done:

call Crlf

call waitMsg

exit

main ENDP

END main

A picture containing shape

Description automatically generated

b)

INCLUDE Irvine32.inc

.data

X dd ?

string1 db "edx: ",0

string2 db "ecx: ",0

string3 db "X: ",0

string4 db "if (edx < ecx) then X=1 else X=2",0

.code

main PROC

lea edx, string4

call writeString

call crlf

lea edx, string1

call writeString

call readDec

mov ebx,eax

lea edx, string2

call writeString

call readDec

mov ecx,eax

mov edx,ebx

cmp edx,ecx

jl smaller

mov X, 2

mov eax,X

lea edx, string3

call writeString

call writeDec

jmp done

smaller:

mov X, 1

mov eax,X

lea edx, string3

call writeString

call writeDec

jmp done

done:

call Crlf

call waitMsg

exit

main ENDP

END main

 Rectangle

Description automatically generated with low confidence

c)

INCLUDE Irvine32.inc

.data

X dd ?

string1 db "edx: ",0

string2 db "ecx: ",0

string3 db "X: ",0

string4 db "if (val1 > ecx)AND (ecx > edx) then X=1 else X=2",0

val1 dd 5

.code

main PROC

lea edx, string4

call writeString

call crlf

lea edx, string1

call writeString

call readDec

mov ebx,eax

lea edx, string2

call writeString

call readDec

mov ecx,eax

mov edx,val1

cmp edx,ecx

jg greater

jmp two

jmp done

greater:

mov edx,ebx

cmp ecx,edx

jl two

mov X, 1

mov eax,X

lea edx, string3

call writeString

call writeDec

jmp done

two:

mov X, 2

mov eax,X

lea edx, string3

call writeString

call writeDec

done:

call Crlf

call waitMsg

exit

main ENDP

END main

A picture containing rectangle

Description automatically generated A picture containing rectangle

Description automatically generated

d)

INCLUDE Irvine32.inc

.data

X dd ?

string1 db "ebx: ",0

string2 db "ecx: ",0

string3 db "X: ",0

string4 db "if (ebx > ecx) OR (ebx > val1) then X=1 else X=2",0

val1 dd 5

.code

main PROC

lea edx, string4

call writeString

call crlf

lea edx, string1

call writeString

call readDec

mov ebx,eax

lea edx, string2

call writeString

call readDec

mov ecx,eax

cmp ebx,ecx

jg one

jl greater

greater:

mov edx,val1

cmp ebx,edx

jl two

mov X, 1

mov eax,X

lea edx, string3

call writeString

call writeDec

jmp done

one:

mov X, 1

mov eax,X

lea edx, string3

call writeString

call writeDec

jmp done

two:

mov X, 2

mov eax,X

lea edx, string3

call writeString

call writeDec

done:

call Crlf

call waitMsg

exit

main ENDP

END main

Rectangle

Description automatically generated with low confidence Rectangle

Description automatically generated with low confidence

e)

INCLUDE Irvine32.inc

.data

X dd ?

string0 db "eax: ",0

string1 db "ebx: ",0

string2 db "ecx: ",0

string3\_5 db "edx: ",0

string3 db "X: ",0

string4 db "if (ebx > ecx AND ebx > edx) OR (ebx > eax) then X=1 else X=2",0

val1 dd ?

.code

main PROC

lea edx, string4

call writeString

call crlf

lea edx, string0

call writeString

call readdec

mov val1,eax

lea edx, string1

call writeString

call readDec

mov ebx,eax

lea edx, string2

call writeString

call readDec

mov ecx,eax

lea edx, string3\_5

call writeString

call readDec

mov edx,eax

cmp ebx,ecx

jg greater

jl lower

greater:

cmp ebx,edx

jg one

jl lower

lower:

mov eax,val1

cmp ebx,eax

jg one

jl two

one:

mov X, 1

mov eax,X

lea edx, string3

call writeString

call writeDec

jmp done

two:

mov X, 2

mov eax,X

lea edx, string3

call writeString

call writeDec

done:

call Crlf

call waitMsg

exit

main ENDP

END main

A screenshot of a computer screen

Description automatically generated A screenshot of a computer screen

Description automatically generated

TASK 2:

INCLUDE Irvine32.inc

.data

array dword 50 DUP(?)

outpur db "Total negative numbers are :",0

comma db ",",0

.code

main PROC

mov ecx,50

mov ebx,0

call randomize

lea esi,array

L1:

call random32

mov [esi],eax

cmp eax,0

jl negtive

jmp down

negtive:

inc ebx

down:

add esi,TYPE Dword

call Writeint

lea edx,comma

call writestring

loop l1;

call crlf

lea edx,outpur

call WriteString

mov eax,ebx

call Writedec

call crlf

call waitmsg

exit

main ENDP

END main

Text

Description automatically generated

TASK 3:

INCLUDE Irvine32.inc

.data

input db "Enter between 0-9 :", 0

wrong db "Invalid input!!",0

string db "\*",0

.code

main PROC

lea edx,input

call WriteString

call readDec

cmp eax,0

jbe invalid

cmp eax,10

jae invalid

mov ecx,eax

call crlf

l1:

lea edx,string

call writestring

loop l1

call crlf

dec eax

mov ecx,eax

cmp eax,0

je oute

ja l1

invalid:

lea edx,wrong

call writeString

oute:

call crlf

call waitmsg

exit

main ENDP

END main



A picture containing rectangle

Description automatically generated

TASK 4:

INCLUDE Irvine32.inc

.data

input db "Enter your score (0-100): ",0

wrong db "Score is invalid !!",0

grade db "Grade :",0

grade\_A db "A+",0

grade\_ac db "A",0

grade\_b db "B",0

grade\_c db "C",0

grade\_f db "F",0

.code

main PROC

lea edx,input

call Writestring

call readdec

A:

cmp eax,100

ja invalid

jbe A\_plus

A\_plus:

cmp eax,90

jb a\_neg

jae A\_pluss

A\_pluss:

lea edx,grade

call Writestring

lea edx,grade\_A

call Writestring

jmp done

a\_neg:

cmp eax,89

jbe b\_neg

b\_neg:

cmp eax,80

jae b\_plus

jbe c\_plus

b\_plus:

lea edx,grade

call Writestring

lea edx,grade\_ac

call Writestring

jmp done

c\_plus:

cmp eax,79

jbe c\_neg

c\_neg:

cmp eax,70

jae c\_print

jbe d\_plus

c\_print:

lea edx,grade

call Writestring

lea edx,grade\_b

call Writestring

jmp done

d\_plus:

cmp eax,69

jbe d\_neg

d\_neg:

cmp eax,60

jae d\_print

jbe fail

d\_print:

lea edx,grade

call Writestring

lea edx,grade\_c

call Writestring

jmp done

fail:

cmp eax,59

jbe fail\_now

fail\_now:

cmp eax,0

jb invalid

jae fail\_print

fail\_print:

lea edx,grade

call Writestring

lea edx,grade\_f

call Writestring

jmp done

invalid:

lea edx,wrong

call writestring

done:

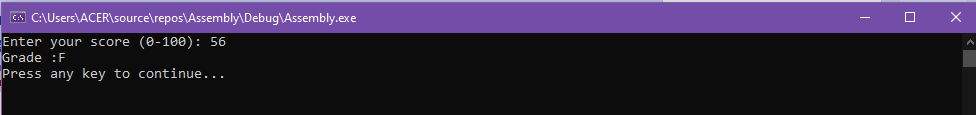
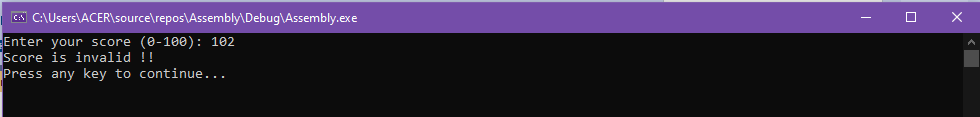
call crlf

call waitmsg

exit

main ENDP

END main

TASK 5:

INCLUDE Irvine32.inc

.data

choice1 db "1: x AND y", 0

choice2 db "2: x OR y", 0

choice3 db "3: NOT x", 0

choice4 db "4: x XOR y", 0

done db "5: EXIT Program", 0

Input db "Enter choice : ", 0

wrong db "Invalid Input !!", 0

Enter\_x db "Enter X: ", 0

Enter\_y db "Enter Y: ", 0

result db "Result: ", 0

dash db "--------------( Operation Complete )----------------", 0

.code

main PROC

Inputloop:

lea edx,choice1

call writeString

call Crlf

lea edx,choice2

call writeString

call Crlf

lea edx,choice3

call writeString

call Crlf

lea edx,choice4

call writeString

call Crlf

lea edx,done

call writeString

call Crlf

lea edx,Input

call writeString

call readInt

cmp eax,1

jb invalid

cmp eax,5

ja invalid

cmp eax,1

jne next

call xANDy

jmp next5

next:

cmp eax,2

jne next2

call xORy

jmp next5

next2:

cmp eax,3

jne next3

call xNOT

jmp next5

next3:

cmp eax,4

jne next4

call xXORy

jmp next5

next4:

cmp eax,5

jne next5

call terminate

next5:

JMP Inputloop

exit

main ENDP

xANDy PROC

lea edx,Enter\_x

call writeString

call readHex

mov ebx,eax

lea edx,Enter\_y

call writeString

call readHex

mov ecx,eax

and ebx,ecx

mov eax,ebx

lea edx,result

call writeString

call writeHex

call Crlf

lea edx,dash

call writeString

call Crlf

ret

xANDy ENDP

xORy PROC

lea edx,Enter\_x

call writeString

call readHex

mov ebx,eax

lea edx,Enter\_y

call writeString

call readHex

mov ecx,eax

or ebx,ecx

mov eax,ebx

lea edx,result

call writeString

call writeHex

call Crlf

lea edx,dash

call writeString

call Crlf

ret

xORy ENDP

xNOT PROC

lea edx,Enter\_x

call writeString

call readHex

not eax

lea edx,result

call writeString

call writeHex

call Crlf

lea edx,dash

call writeString

call Crlf

ret

xNOT ENDP

xXORy PROC

lea edx,Enter\_x

call writeString

call readHex

mov ebx,eax

lea edx,Enter\_y

call writeString

call readHex

mov ecx,eax

xor ebx,ecx

mov eax,ebx

lea edx,result

call writeString

call writeHex

call Crlf

lea edx,dash

call writeString

call Crlf

ret

xXORy ENDP

invalid PROC

lea edx,wrong

call writeString

call Crlf

ret

invalid ENDP

terminate PROC

call Crlf

call Waitmsg

terminate ENDP

END main Text

Description automatically generated Text

Description automatically generated

TASK 6:

INCLUDE Irvine32.inc

.data

inputS db "Please ener a string: ", 0

output1 db "String is palindrome!", 0

output2 db "String is not palindrome!", 0

string dd 50 DUP(?)

reverse dd 50 DUP(?)

.code

main PROC

call input

call checkPalindrome

call crlf

call waitmsg

exit

main ENDP

input PROC

lea edx,inputS

call writeString

lea edx,string

mov ecx, SIZEOF string

call readString

ret

input ENDP

checkPalindrome PROC

mov edx,eax

lea esi,string

lea edi,reverse

dec dl

add esi,edx

push eax

mov ecx,eax

LOOP1:

push [esi]

pop eax

mov [edi],eax

add edi,1

sub esi,1

LOOP LOOP1

lea esi,string

lea edi,reverse

pop eax

mov ecx,eax

LOOP2:

mov eax,[esi]

mov edx,[edi]

mov ebx,0

mov bl,al

mov eax,0

mov eax,ebx

mov ebx,0

mov bl,dl

mov edx,0

mov edx,ebx

cmp edx,eax

JNE notPalindrome

add edi,1

add esi,1

LOOP LOOP2

jmp isPalindrome

notPalindrome:

lea edx,output2

call writeString

call Crlf

ret

isPalindrome:

lea edx,output1

call writeString

call Crlf

ret

checkPalindrome ENDP

END main

