LAB 08

Q1)

include irvine32.inc

.data

msg BYTE "Equal",0

msg2 BYTE "NOT EQUAL",0

a BYTE ?

b BYTE ?

f BYTE ?

d BYTE ?

.code

main PROC

call readint

mov edx,eax

call readint

mov esi,eax

call readint

mov ecx,eax

call readint

cmp edx,esi

JE L1

mov edx, offset msg2

call writeString

exit

L1:

CMP esi,ecx

JE L2

mov edx, offset msg2

call writeString

exit

L2:

CMP ecx,eax

JE L3

mov edx,offset msg2

call writeString

exit

L3:

mov edx, offset msg

call writeString

exit

main ENDP

end main

Q2)

include irvine32.inc

.data

intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0

msg BYTE "First zero found at: ",0

.code

main PROC

mov eax,0

mov ecx,LENGTHOF intArr

mov esi,-1

L1:

inc esi

CMP ax,intArr[esi]

JE L2

LOOP L1

inc esi

CMP esi,ecx

JMP L3

L2:

mov edx,offset msg

call writestring

mov eax,esi

call writedec

L3:

exit

main ENDP

end main

Q3)

include irvine32.inc

.data

intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0

var DWORD 5

X DWORD ?

msg BYTE "X: ",0

.code

main PROC

mov edx,var

inc edx

mov ecx,LENGTHOF intArr

CMP var,ecx

JL L1

mov x,1

mov eax,x

exit

L1:

CMP ecx,edx

JGE L2

mov x,1

mov eax,x

call writedec

exit

L2:

mov x,0

mov eax,0

call writedec

exit

main ENDP

end main

Q4)

include irvine32.inc

.data

var DWORD 0

msg BYTE "Hello",0

msg2 BYTE "World",0

.code

main PROC

L1:

CMP var,5

JL L2

mov edx,offset msg2

call writeString

add var,1

CMP var,10

JLE L1

exit

L2:

mov edx,offset msg

call writestring

add var,1

JMP L1

L3:

exit

main ENDP

end main

Q5)

include irvine32.inc

.data

arr word 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20

msg BYTE "Enter the value for sequential search: ",0

msg2 BYTE "Value found at: ";

.code

main PROC

mov edx,offset msg

call writeString

call readint

mov ecx,LENGTHOF arr

mov esi,-1

L1:

inc esi

cmp ax,arr[esi]

JE L2

LOOP L1

exit

L2:

mov edx,offset msg2

call writestring

mov eax,esi

call writedec

exit

main ENDP

end main

Q6)

Include Irvine32.inc

.data

swapCount BYTE 0

list BYTE 1,2,3,4,5,6,7,8,9,10

.code

main proc

mov ecx,LENGTHOF list

mov esi,0

L1:

movsx eax,list[esi]

movsx ebx,list[esi+1]

inc esi

CMP eax,ebx

JG L2

L3:

LOOP L1

cmp ecx,0

je l4

L2:

movsx eax,list[esi]

movsx edx,list[esi+1]

add swapCount,1

JMP L3

l4:

movsx eax,swapCount

call writedec

call crlf

exit

main endp

**Q7**

INCLUDE Irvine32.inc

.data

day Dword ?

DAY\_1 byte "MONDAY",0 DAY\_2 byte "TUESDAY",0 DAY\_3 byte "WEDNESDAY",0 DAY\_4 byte "THURSDAY",0

DAY\_5 byte "FRIDAY",0 DAY\_6 byte "SATURDAY",0 DAY\_7 byte "SUNDAY",0

NO\_DAY BYTE "Wrond day inserted",0

.code

main PROC call readint mov DAY,eax cmp eax,1

je DAY1 cmp eax,2 je DAY2 cmp eax,3 je DAY3 cmp eax,4 je DAY4 cmp eax,5 je DAY5 cmp eax,6 je DAY6 cmp eax,7

je DAY7 jne NODAY

DAY1:

call crlf

mov edx,OFFSET DAY\_1 call writestring

exit DAY2:

call crlf

mov edx,OFFSET DAY\_2 call writestring

exit DAY3:

call crlf

mov edx,OFFSET DAY\_3 call writestring

exit DAY4:

call crlf

mov edx,OFFSET DAY\_4

call writestring exit

DAY5:

call crlf

mov edx,OFFSET DAY\_5 call writestring

exit DAY6:

call crlf

mov edx,OFFSET DAY\_6 call writestring

exit DAY7:

call crlf

mov edx,OFFSET DAY\_7 call writestring

exit NODAY:

call crlf

mov edx,OFFSET NO\_DAY call writestring

exit

main ENDP END main

**Q8**

INCLUDE Irvine32.inc

.data

day Dword ?

DAY\_1 byte " : IS AN ALPHABET",0 DAY\_2 byte " : NOT AN ALPHABET",0

.code

main PROC call readchar cmp al,65

je truee jc falsee cmp al,90 je truee

jc truee

cmp al,97 je truee jc falsee

cmp al,122 je truee

jc truee jnc falsee

truee:

call writechar

mov edx,OFFSET DAY\_1 call writestring

exit

falsee:

call writechar

mov edx,OFFSET DAY\_2 call writestring

exit

exit

exit

main ENDP END main