COAL-LAB 8

**TASK 1:**

INCLUDE Irvine32.inc

.data

msg1 byte "Integers Found Equal to number ",0

msg2 byte "No Integer is Equal!",0

sign byte " = ",0

integers dword ?

larger dword ?

x dword ?

y dword ?

z dword ?

w dword ?

.code

main PROC

mov w,0

mov esi, offset integers

mov z,esi

mov ecx,4

input:

call readint

mov [esi],eax

add esi,4

loop input

mov esi,z

mov ecx,4

mov edi, offset integers

call crlf

mov z,4

compare:

cmp w, 4

je fin

mov eax,[edi]

mov ecx,z

cmp z,0

je fin

compare2:

add esi,4

mov ebx,[esi]

mov x,eax

cmp eax, ebx

je cond

call crlf

mov edx,offset msg2

call writestring

loop compare2

add edi,4

sub z,1

loop compare

cond:

add edi,4

add w,1

cmp w,1

jg equal

jmp compare2

loop cond

equal:

call crlf

mov eax,w

call writeint

mov edx,offset msg1

call writestring

mov eax,x

call writeint

add esi,4

jmp compare

cmp w, 4

je fin

loop equal

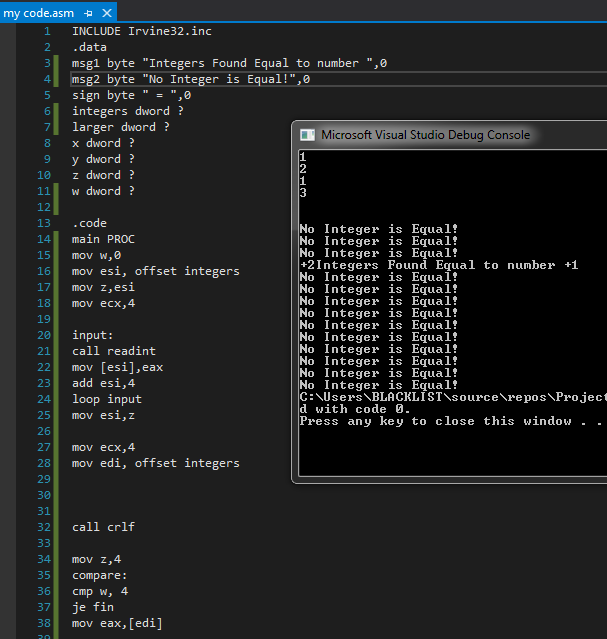
fin:

exit

loop fin

main ENDP

END main



**TASK 2:**

INCLUDE Irvine32.inc

.data

msg1 byte "First Non Zero value Found = ",0

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

.code

main PROC

mov ecx, lengthof intArr

mov esi,offset intArr

nonzero:

mov eax,[esi]

mov edx,0

mov ebx,10

div ebx

cmp edx, 0

jne fin

add esi, 2

loop nonzero

fin:

mov edx,offset msg1

call writestring

mov ax, [esi]

movsx eax,ax

call writeint

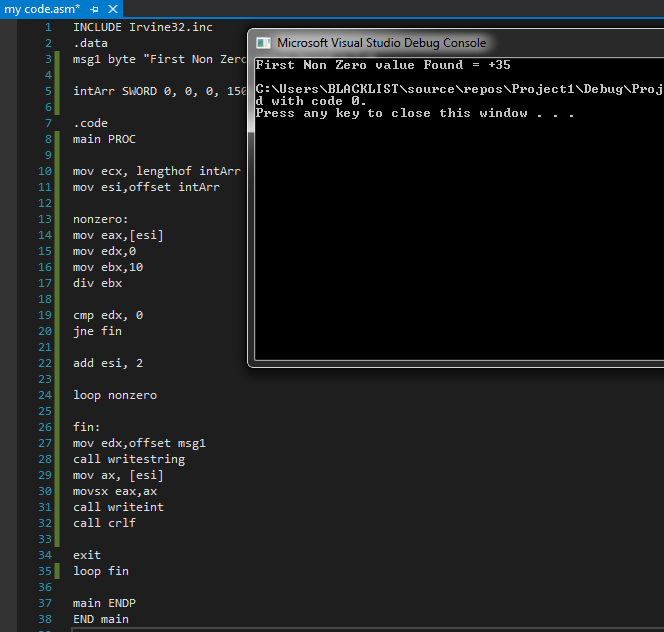
call crlf

exit

loop fin

main ENDP

END main



**TASK 3:**

INCLUDE Irvine32.inc

.data

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

var dword ?

x dword ?

y dword ?

.code

main PROC

mov var,5

mov ecx,lengthof intArr

mov ebx,ecx

mov y, -1

l1:

cmp y,ecx

jge fin

add y,1

call crlf

add var,1

mov edx,var

cmp ecx,0

je fin

cmp var,ecx

jl l2

l2:

cmp ecx,edx

jge l3

mov x,1

call else1

l3:

mov x,0

mov eax,x

call writedec

call crlf

jmp l1

loop l3

loop l1

else1:

mov eax,x

call writedec

call crlf

jmp l1

loop else1

fin:

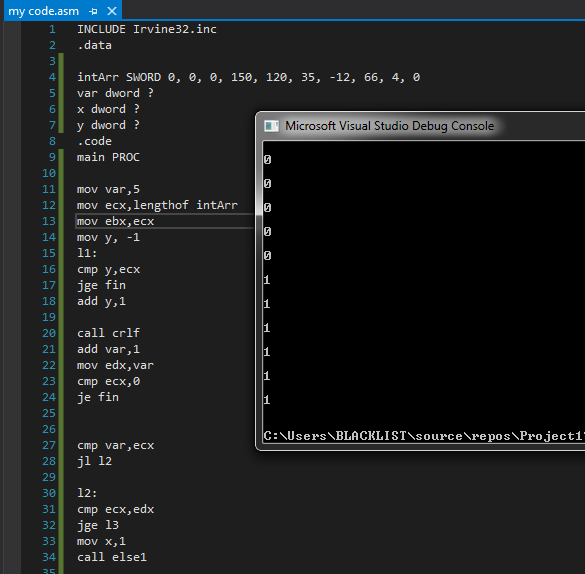
call crlf

exit

loop fin

main ENDP

END main



**TASK 4:**

INCLUDE Irvine32.inc

.data

msg1 byte " -HELLO-",0

msg2 byte " -WORLD- ",0

var dword ?

.code

main PROC

mov var,-1

l1:

add var,1 ; Here, var=0 on first iteration

cmp var , 10

jg fin

mov eax,var

call writedec

cmp var ,5

jl hello

cmp var ,5

jge world

loop l1

hello:

mov edx, offset msg1

call writestring

call crlf

call crlf

jmp l1

world:

mov edx, offset msg2

call writestring

call crlf

call crlf

jmp l1

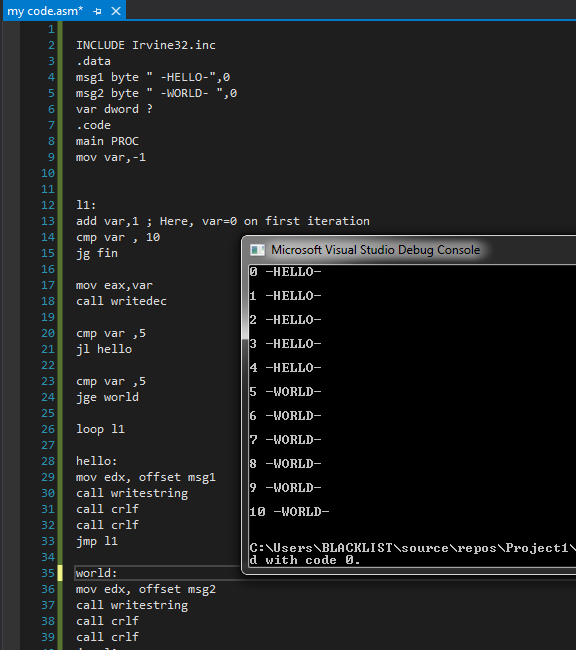
fin:

exit

loop fin

main ENDP

END main



**TASK 5:**

INCLUDE Irvine32.inc

.data

msg1 byte "Please enter an Integer = ",0

msg2 byte "Integer match Found !",0

msg3 byte "No Integer match Found !",0

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

x dword ?

y dword ?

.code

main PROC

mov edx ,offset msg1

call writestring

call readint

mov x,eax

mov ecx, lengthof arr

mov esi, offset arr

mov y,esi

call crlf

display:

mov ax,[esi]

movzx eax,ax

call writedec

call crlf

add esi,2

loop display

mov eax,x

mov esi,y

mov ecx, lengthof arr

check:

mov bx,[esi]

movzx ebx,bx

cmp ax,bx

je found

add esi,2

loop check

jmp notfound

found:

call crlf

mov edx , offset msg2

call writestring

call crlf

exit

loop found

notfound:

call crlf

mov edx , offset msg3

call writestring

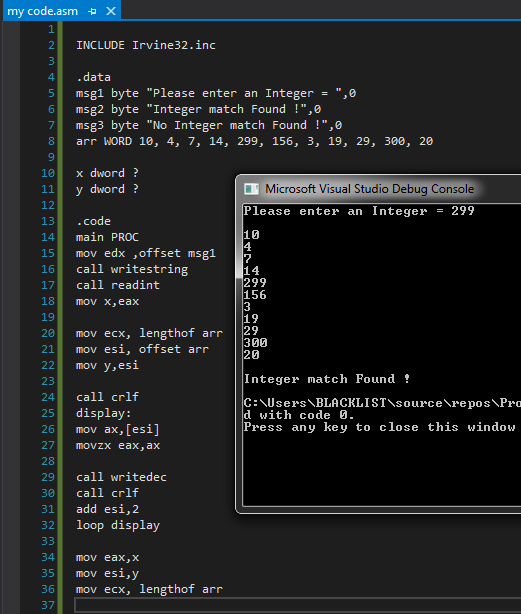
call crlf

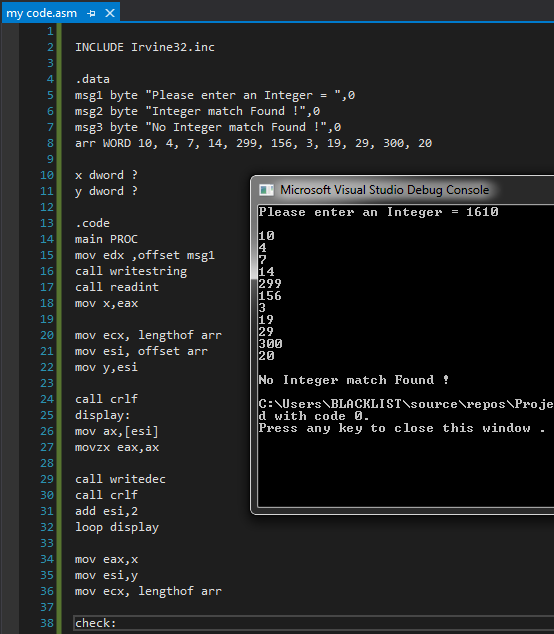
exit

loop notfound

main ENDP

END main





**TASK 6:**

INCLUDE Irvine32.inc

.data

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

i DWORD 1

temp DWORD ?

.code

main PROC

mov ecx, lengthof arr

l1:

mov temp, ecx

mov eax, 0

mov ebx, 0

sub ecx, i

mov esi, 0

mov edi, 1

l2:

mov ax, arr[esi\*TYPE arr]

mov bx, arr[edi\*TYPE arr]

cmp ax, bx

JG exchange

JLE cont

exchange:

mov ax, arr[esi\*TYPE arr]

mov bx, arr[edi\*TYPE arr]

xchg ax, arr[edi\*TYPE arr]

xchg bx, arr[esi\*TYPE arr]

cont:

inc esi

inc edi

cmp ecx, 0

JLE outsideinner

loop l2

outsideinner:

mov ecx, temp

inc i

loop l1

mov edi, 0

mov ecx, LENGTHOF arr

mov esi, 0

l3:

mov ax, arr[esi\*TYPE arr]

inc esi

call writedec

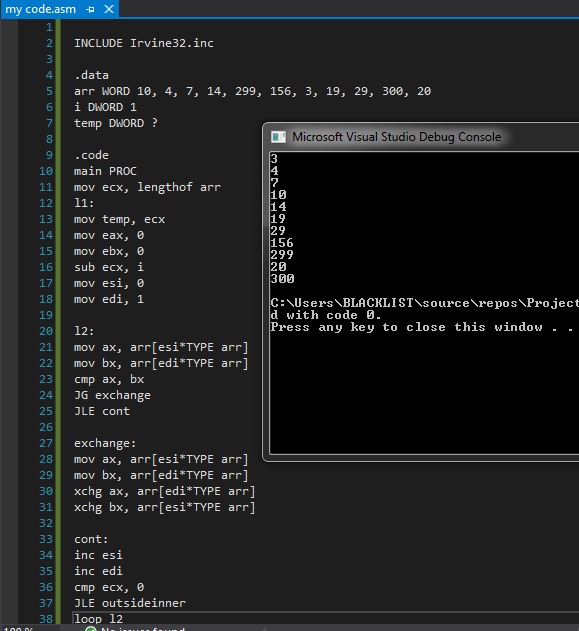
call crlf

loop l3

exit

main ENDP

END main



**TASK 7:**

INCLUDE Irvine32.inc

.data

week byte " ~Week Days With Numeric Order~",0

msg byte "Enter a Number between 1 - 7 : ",0

mon byte " MONDAY",0

tue byte " TUESDAY",0

wed byte " WEDNESDAY",0

thurs byte " THURSDAY",0

fri byte " FRIDAY",0

sat byte " SATURDAY",0

sun byte " SUNDAY",0

invalid byte " Invalid entry! Please enter valid week-day number 1 - 7!",0

.code

main PROC

call crlf

mov edx, offset week

call writestring

call crlf

num:

call crlf

mov edx, offset msg

call writestring

call readdec

cmp eax,1

jl fin

cmp eax,1

je mo

cmp eax,2

je tu

cmp eax,3

je we

cmp eax,4

je th

cmp eax,5

je fr

cmp eax,6

je sa

cmp eax,7

je su

cmp eax,7

jge fin

loop num

mo:

mov edx, offset mon

call writestring

call crlf

jmp num

tu:

mov edx, offset tue

call writestring

call crlf

jmp num

we:

mov edx, offset wed

call writestring

call crlf

jmp num

th:

mov edx, offset thurs

call writestring

call crlf

jmp num

fr:

mov edx, offset fri

call writestring

call crlf

jmp num

sa:

mov edx, offset sat

call writestring

call crlf

jmp num

su:

mov edx, offset sun

call writestring

call crlf

jmp num

fin:

call crlf

mov edx, offset invalid

call writestring

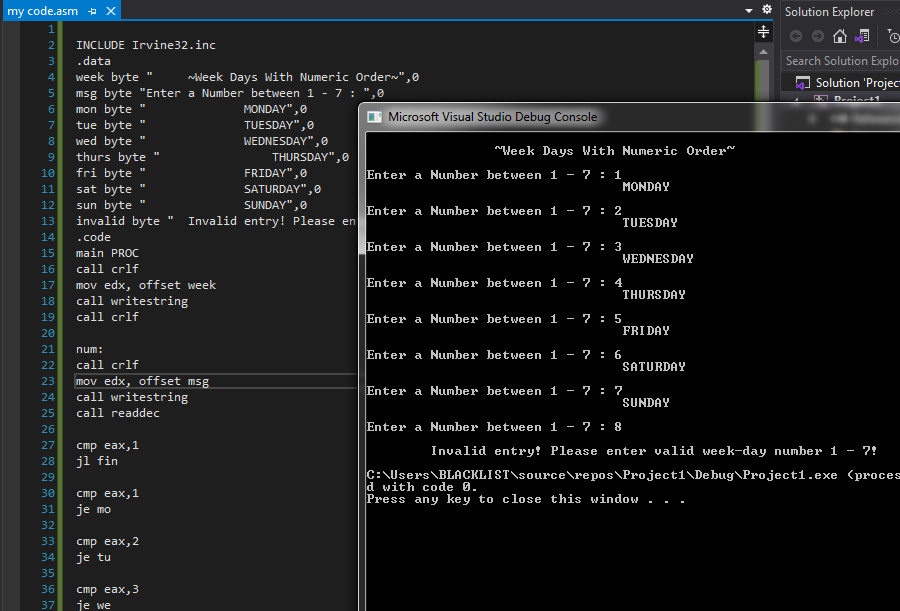
call crlf

exit

loop fin

main ENDP

END main



**TASK 8:**

INCLUDE Irvine32.inc

.data

msg byte "Enter a character : ",0

msg2 byte "Character is an Alphabet!",0

msg3 byte "Character is not an Alphabet!",0

letter byte 65,0

.code

main PROC

call crlf

mov edx, offset msg

call writestring

call readchar

call writechar

call crlf

mov ecx ,255

mov edx,offset letter

compare:

cmp al,letter

je ALPHA

inc letter

loop compare

call crlf

mov edx, offset msg3

call writestring

call crlf

exit

ALPHA:

call crlf

mov edx, offset msg2

call writestring

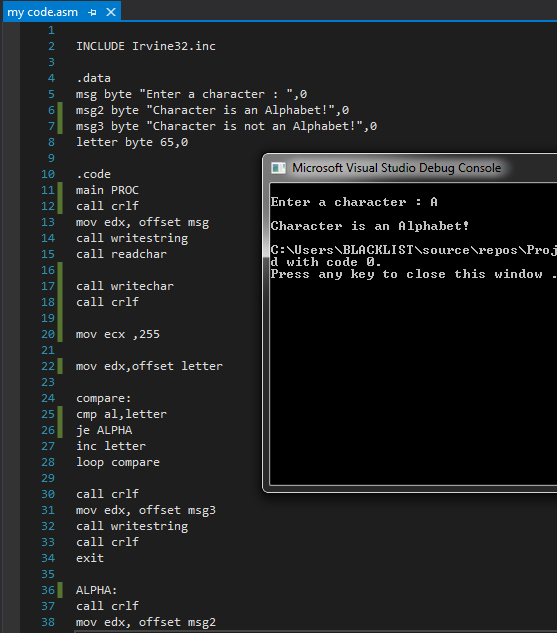
call crlf

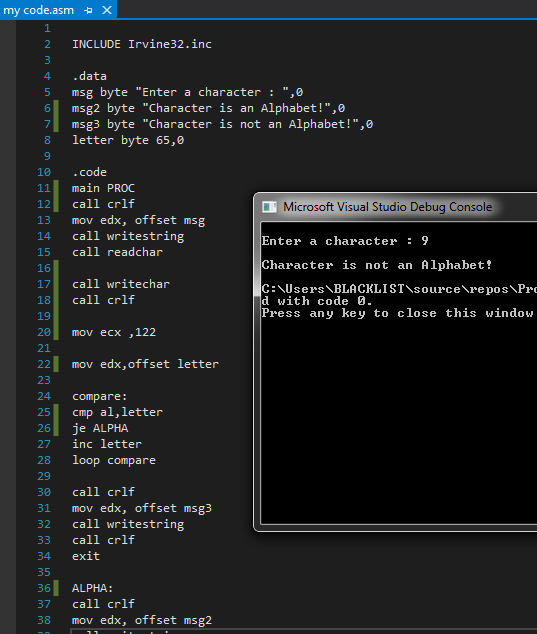
exit

loop ALPHA

main ENDP

END main





NAME: MAQBOOL AHMED

ROLLNO: 20K-1610