**Name: Ibadullah**

**Roll No: 19k-0259**

**Lab 11 Tasks.**

**------------------------------------------------------------------------------------**

**Task 1:**

**Code:**

INCLUDE Irvine32.inc

func PROTO, start: DWORD, ending: DWORD, cnt: DWORD

.data

count DWORD 0

msg1 BYTE "Enter Starting Index: ",0

msg2 BYTE "Enter Ending Index: ",0

msg3 BYTE "Factors of 3 are: ",0

msg4 BYTE "Count: ",0

one DWORD ?

endin DWORD ?

.code

main PROC

mov eax, 0

mov edx, OFFSET msg1

call WriteString

call readInt

mov one, eax

mov eax, 0

mov edx, OFFSET msg2

call WriteString

call readInt

mov endin, eax

mov eax, 0

mov edx, OFFSET msg3

call WriteString

INVOKE func, one, endin, count

exit

main ENDP

func PROC, start: DWORD, ending: DWORD, cnt: DWORD

mov eax, start

mov ebx, ending

sub ebx, eax

mov eax, start

mov ecx, ebx

INC ecx

L1:

mov edx, 0

mov ebx, 3

PUSH eax

div ebx

cmp edx, 0

je done

POP eax

jne not\_done

done:

POP eax

call WriteDec

PUSH eax

mov eax, ","

call WriteChar

POP eax

INC cnt

not\_done:

INC eax

Loop L1

call crlf

mov edx, OFFSET msg4

call WriteString

mov eax, cnt

call WriteDec

call crlf

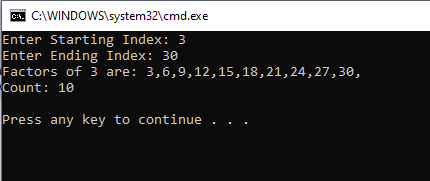
call crlf

RET

func ENDP

END main

**Screenshot:**



**Task 2:**

**Code:**

INCLUDE Irvine32.inc

get\_cal PROTO, one: DWORD, two: DWORD

.data

msg1 BYTE "Enter Marks of 10 Students: ",0

msg2 BYTE "Average of odd idexes is: ",0

msg3 BYTE "Marks: ",0

avg DWORD 0

array DWORD 10 DUP (?)

.code

main PROC

mov eax, 0

mov ebx, 0

INVOKE get\_cal, ADDR array, avg

exit

main ENDP

get\_cal PROC one: DWORD, two: DWORD

mov edx, OFFSET msg1

call WriteString

mov esi, 0

mov ecx, LENGTHOF array

L1:

call ReadDec

mov array[esi\* TYPE array], eax

INC esi

Loop L1

call clrscr

mov edx, OFFSET msg3

call WriteString

mov esi, 0

mov ecx, LENGTHOF array

l2:

mov eax, array[esi\*TYPE array]

call writeDec

mov eax, ","

call WriteChar

INC esi

Loop l2

mov ecx, LENGTHOF array

mov esi, 0

l3:

mov eax, array[esi\*TYPE array]

add two, eax

add esi, 2

Loop l3

call crlf

mov eax, two

mov edx, 0

mov ebx, 5

div ebx

mov edx, OFFSET msg2

call WriteString

call writeDec

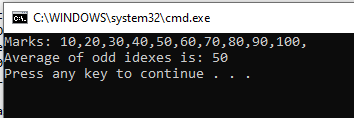
call crlf

RET

get\_cal ENDP

END main

**Screenshot:**



**Task 3:**

**Code:**

INCLUDE Irvine32.inc

.data

msg1 BYTE "Enter Base Number: ",0

msg2 BYTE "Enter Power Number: ",0

msg3 BYTE "Answer is Odd.",0

msg4 BYTE "Answer is Even.",0

msg5 BYTE "Base Number to the Power Number is: ",0

base DWORD ?

power DWORD ?

res DWORD 1

.code

main PROC

mov edx, OFFSET msg1

call writeString

call ReadInt

mov base, eax

mov edx, OFFSET msg2

call writeString

call ReadInt

mov power, eax

mov ecx, power

mov eax, base

call Pow

mov edx, OFFSET msg5

call writeString

call WriteDec

call crlf

mov edx, 0

mov ebx, 2

div ebx

cmp edx, 0

je ev

mov edx, OFFSET msg3

call writeString

call crlf

jmp akhir

ev:

mov edx, OFFSET msg4

call writeString

call crlf

akhir:

exit

main ENDP

Pow PROC

cmp ecx, 1

je equal

mov ebx, base

mul ebx

DEC ecx

call Pow

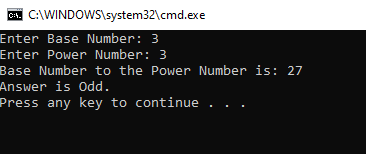
equal:

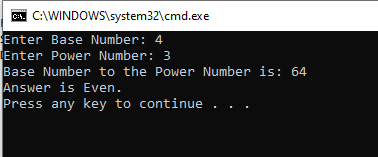
RET

Pow ENDP

END main

**Screenshot:**





**Task 4 is same as task 2…!!**

**Thankyou. ☺**