**COAL LAB 7**

Questions:

Task#1:

Take an array of 10 numbers move word-type of data into another empty array using stack push

and pop technique.

Task#2

Write a program which displays the addition of three integers through a stack.

Task#3

Write a program having procedures are used to calculate the total sum of 2 arrays (each array

having 5-elements). The sum of 1-array in 1st procedure and in 2nd procedure have sum of 2-

array. And the 3rd procedure add the results of both.

Task#4

Print the following pattern using a function call in which number of columns is pass through a

variable.

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

Task#5

Write a function that asks the user for a number n and prints the sum of the numbers 1 to n

Q1

INCLUDE Irvine32.inc

.data

str1 BYTE "Enter the number: " , 0

arr WORD 10 DUP(?) ; array of 10 elements

arr2 WORD 10 DUP(?) ;

.code

main PROC

;taking input

mov eax , 0

mov esi , 0

mov ecx , 10

input:

mov edx , offset str1

call Writestring

call Readint

mov arr[esi] , ax

add esi , 2

loop input

mov eax , 0

mov esi , 0

mov ecx , 10

transfer:

mov bx , arr[esi]

push bx ; value pushed in stack

pop dx

mov arr2[esi] , dx

add esi , 2

loop transfer

;printing data

mov ecx , 10

mov eax , 0

mov esi , 0

print:

mov ax , arr2[esi]

call Writeint

call crlf

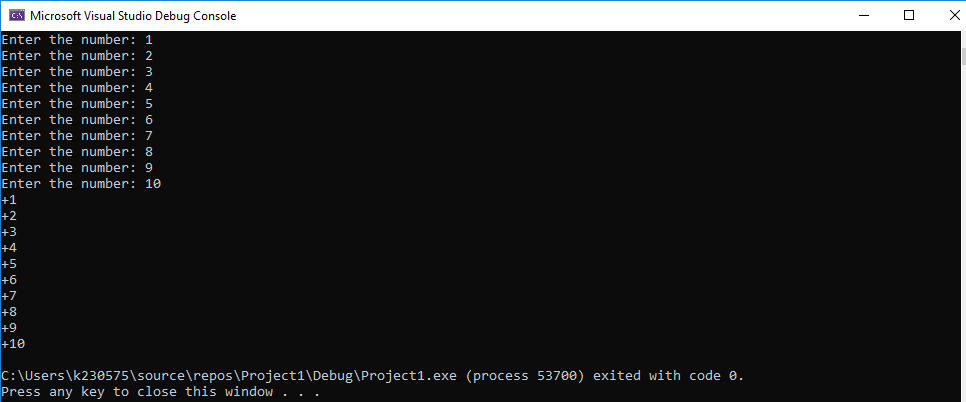
add esi , 2

loop print

exit

main ENDP

END main



Q2

Code:

INCLUDE Irvine32.inc

.data

var1 DWORD 10

var2 DWORD 5

var3 DWORD 20

.code

main PROC

;pushing the values in the stack

mov eax , 0

mov eax , var1

push eax

mov ebx ,var2

push ebx

mov ecx , var3

push ecx

;popping the values out of stack and adding it

mov eax , 0

mov ebx , 0

mov ecx , 0

pop eax

pop ebx

add eax , ebx

pop ecx

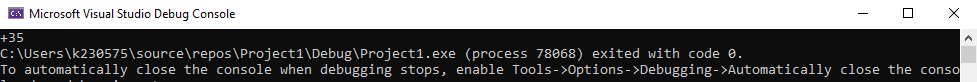
add eax , ecx

call Writeint

exit

main ENDP

END main



Q3

INCLUDE Irvine32.inc

.data

arr1 DWORD 1 , 2 , 3 , 4 , 5

arr2 DWORD 6 , 7 , 8 , 9 , 10

sum1 DWORD ?

sum2 DWORD ?

.code

main PROC

call Arr1Sum

call Arr2Sum

call Sum

exit

;procedure to calculate the sum of first array

Arr1Sum PROC

mov eax , 0

mov esi , 0

mov ecx , 5

L1:

add eax , arr1[esi \*TYPE arr1]

inc esi

loop L1

;call Writeint

mov sum1 , eax

ret

Arr1Sum ENDP

;procedure to calculate the sum of second array

Arr2Sum PROC

mov eax , 0

mov esi , 0

mov ecx , 0

mov ecx , 5

L2:

add eax , arr2[esi \*TYPE arr2]

inc esi

loop L2

mov sum2 , eax

ret

Arr2Sum ENDP

;procedure to add the sum of both the arrays

Sum PROC

mov eax , 0

mov eax , sum1

add eax , sum2

call Writeint ; printing the sum

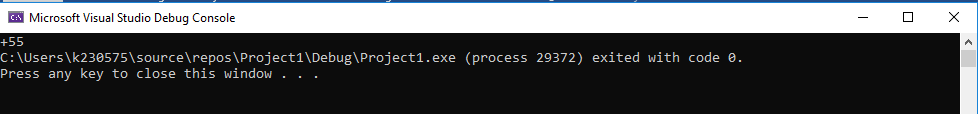
ret

Sum ENDP

;exit

main ENDP

END main



Q4

Code:

INCLUDE Irvine32.inc

.data

spaces DWORD 5

rows DWORD ?

cols DWORD 1

.code

main PROC

call printPattern

exit

main ENDP

printPattern PROC

mov ecx,5

L1:

mov rows,ecx

mov ecx,spaces

L2:

mov al," "

call writechar

LOOP L2

dec spaces

mov ecx,cols

L3:

mov al,"\*"

call writechar

LOOP L3

inc cols

call crlf

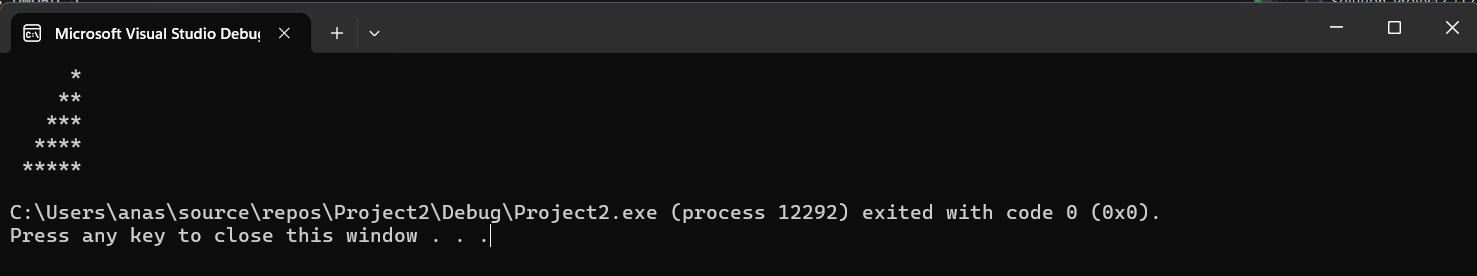
mov ecx,rows

LOOP L1

ret

printPattern ENDP

END main



Q5

Code:

INCLUDE Irvine32.inc

.data

str1 BYTE "Enter the number n: " , 0

n DWORD ?

count DWORD ?

.code

main PROC

mov edx , offset str1

call Writestring

call Readint

mov n , eax

call SumN

exit

SumN PROC

mov eax , 0

mov ecx , n

mov eax , n

L1:

dec n

add eax , n

loop L1

call Writeint

ret

SumN ENDP

main ENDP

END main

