**ASSEMBLY FUNDAMENTALS 2**

**LAB TASK**

**NIMRA JAMIL 10187**

**QUESTION 1:**

**CODE :**

Include Irvine32.inc

.data

QUESTION1 BYTE "QUESTION 1 : DIVIDING 2 NUMBERS AND PRINTING QUOTIENT AND REMAINDER. ",0dh,0ah,0

QUOTIENT BYTE "QUOTIENT : ",0dh,0ah,0

REMAINDER BYTE "REMAINDER : ",0dh,0ah,0

.code

main PROC

mov eax,6

mov ebx,3

SUB EDX,EDX

div ebx

MOV EDX,OFFSET QUOTIENT

CALL WRITESTRING

Call writeDEC

call crlf

exit

main ENDP

END main



**QUESTION 2 PART A) :**

**CODE :**

Include Irvine32.inc

.data

QUESTION BYTE "QUESTION 2 : PART A) (5+2)\*(6-4) ",0dh,0ah,0

ANSWER BYTE "ANSWER : "

.code

main PROC

MOV EDX,OFFSET QUESTION

CALL WRITESTRING

mov eax,5

mov ebx,2

add eax,ebx

mov ecx,eax

mov eax,6

mov ebx,4

sub eax,ebx

imul eax,ecx

MOV EDX,OFFSET ANSWER

CALL WRITESTRING

Call writeDEC

call crlf

exit

main ENDP

END main



**QUESTION 2 PART B):**

**CODE :**

Include Irvine32.inc

.data

QUESTION BYTE "QUESTION 2 : PART B) (7\*8)-(3\*2) ",0dh,0ah,0

ANSWER BYTE "ANSWER : "

.code

main PROC

MOV EDX,OFFSET QUESTION

CALL WRITESTRING

mov edi,7

mov esi,8

imul edi,esi

mov esi,3

mov ebx,2

imul esi,ebx

sub esi,edi

MOV EDX,OFFSET ANSWER

CALL WRITESTRING

Call writeDEC

call crlf

exit

main ENDP

END main



**QUESTION 2 PART C):**

**CODE :**

Include Irvine32.inc

.data

QUESTION BYTE "QUESTION 2 : PART B) (4/2)\*(4 + 4-2)/2 ",0dh,0ah,0

ANSWER BYTE "ANSWER : "

.code

main PROC

MOV EDX,OFFSET QUESTION

CALL WRITESTRING

mov eax,4

mov ebx,2

sub edx,edx

div ebx

mov esi,eax

mov eax,4

mov ebx,4

add eax,ebx

mov ebx,2

sub eax,ebx

sub edx,edx

div eax

mul esi

MOV EDX,OFFSET ANSWER

CALL WRITESTRING

Call writeDEC

call crlf

exit

main ENDP

END main



**QUESTION 2 PART D):**

**CODE :**

Include Irvine32.inc

.data

QUESTION BYTE "QUESTION 2 : PART B) (2\*2)/(2\*3\*4)-(5+4-1) ",0dh,0ah,0

ANSWER BYTE "ANSWER : "

.code

main PROC

MOV EDX,OFFSET QUESTION

CALL WRITESTRING

mov edi,2

mov ebx,2

imul edi,ebx

mov ecx,3

imul ecx,ebx

mov ebx,4

imul ecx,ebx

mov esi,5

mov ebx,4

add esi,ebx

mov ebx,2

sub esi,ebx

sub ecx,esi

mov eax,ecx

sub edx,edx

div esi

MOV EDX,OFFSET ANSWER

CALL WRITESTRING

Call writeDEC

call crlf

exit

main ENDP

END main

