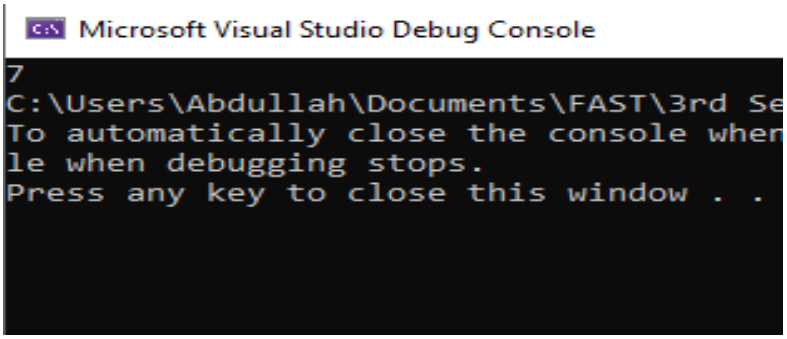
COAL LAB 11

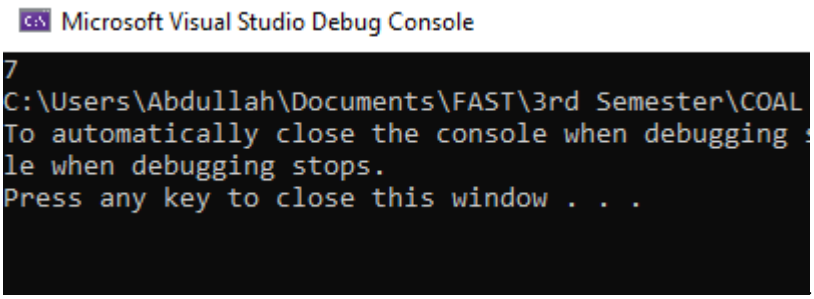
Question 1

```
Include Irvine32.inc

.data
	Str1 BYTE "127&j~3#^&*#*#45^",0

.code
	Main PROC
		call Scan_String
		MOV eax, ebx
		call writedec

		exit
		main endp

	Scan_String PROC
		MOV edi, offset str1
		MOV ecx, lengthof str1
		MOV ebx, 0
		MOV eax, 0
		MOV al, '#'
		L1:
			cmp [edi], al
			JE endd
			inc ebx
			inc edi
		loop L1

		endd:
		ret
		Scan_String endp
	end main
```



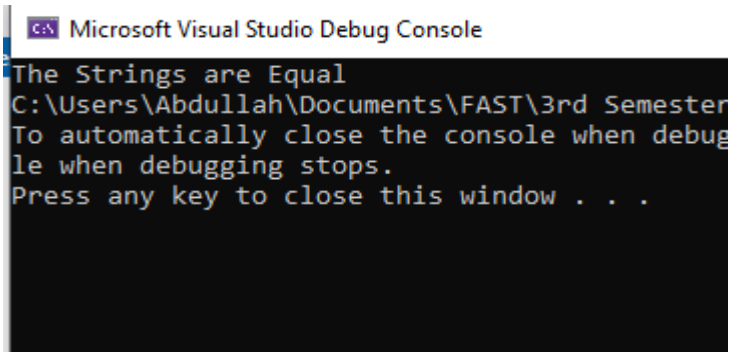
Question 2

```
Include Irvine32.inc

Scan_String PROTO, ptrstr1:DWORD, lenstr:DWORD, charsr:BYTE

.data
	Str1 BYTE "127&j~3#^&*#*#45^",0

.code
	Main PROC
		INVOKE Scan_String, ADDR str1, lengthof str1, '#'
		MOV eax, ebx
		call writedec

		exit
		main endp

	Scan_String PROC, ptrstr1:DWORD, lenstr:DWORD, charsr:BYTE
		MOV edi, ptrstr1
		MOV ecx, lenstr
		MOV ebx, 0
		MOV eax, 0
		MOV al, charsr
		L1:
			cmp [edi], al
			JE endd
			inc ebx
			inc edi
		loop L1

		endd:
		ret
		Scan_String endp
	end main
```



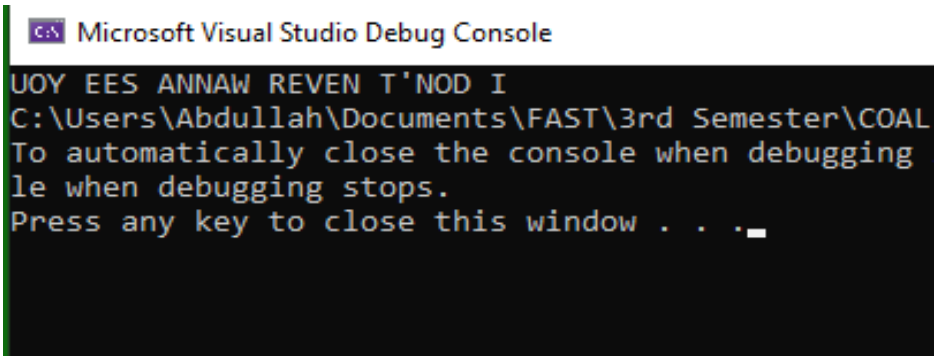
Question 3

```
Include Irvine32.inc

.data
	str1 BYTE "The Strings are Equal", 0
	str2 BYTE "The Strings are not Equal", 0
	str3 BYTE "WHAT IF I TOLD YOU THAT I LOVE YOU?", 0
	str4 BYTE "WHAT IF I TOLD YOU THAT I LOVE YOU?", 0

.code
	Main PROC
		push offset str3
		push offset str4
		call Iscompare

		exit
		main endp

	Iscompare PROC
		LOCAL check:BYTE

		cld
		MOV esi, [ebp+8]
		MOV edi, [ebp+12]
		MOV ecx, [ebp+16]

		cmpsb
		JNZ endd
		MOV edx, offset str1
		call writestring
		jmp enddd

		endd:
			MOV edx, offset str2
			call writestring

		enddd:
		ret
		Iscompare endp
	end main
```



Question 4

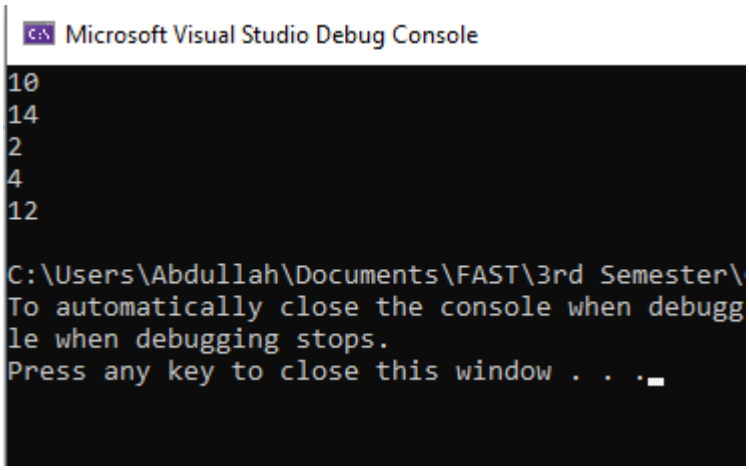
```
Include Irvine32.inc

.data
	str1 BYTE "I DON'T NEVER WANNA SEE YOU", 0

.code
	Main PROC
		MOV eax, 0
		MOV ecx, lengthof str1
		call Str_Reverse

		exit
		Main endp

	Str_Reverse PROC
		MOV al, [str1+ecx]
		call writechar
		dec ecx
		cmp ecx, 0
		JL endd
		call Str_Reverse

		endd:
		ret
		Str_Reverse endp

	end main
```



Question 5

```
Include Irvine32.inc

.data
        arr1 DWORD 5, 7, 1, 2, 6
        byteno DWORD 2

.code
        main PROC
                push offset arr1
                MOV ecx, lengthof arr1
                push ecx
                push byteno

                call multiply

                MOV ecx, lengthof arr1
                MOV esi, offset arr1
                PRINT:
                        MOV eax, [esi]
                        call writedec
                        call crlf
                        ADD esi, 4
                        loop PRINT

                exit
                main endp

        multiply PROC
                cld
                enter 0,0
                MOV esi, [ebp+16]
                MOV edi, esi
                MOV ecx, [ebp+12]
                L1:
```

```
        LODSD
        mul byteno
        STOSD
        loop L1


    leave
    ret
    multiply endp

end main
```



Question 6

```
INCLUDE Irvine32.inc

Get_frequencies PROTO, targ: PTR dword, freq: PTR dword

.data
	target BYTE "AAEBDCFBBC",0
	freqTable DWORD 256 DUP(0)
	eql byte '=',0
	var byte 65d


.code
	main PROC
		mov esi, offset target
		mov esi, offset freqTable

		INVOKE Get_frequencies, ADDR target, ADDR freqTable

		movzx eax, target[3]
		mov ebx, 4
		imul ebx
		mov ebx, eax
		mov eax, freqTable[ebx]
		call crlf
		mov esi,OFFSET freqTable
		mov ecx,LENGTHOF freqTable
		mov bl,00

		l1:
			mov al,bl
			cmp al,var
			jne L2
			call writechar
			mov eax, [esi]
			mov edx,OFFSET eql
			call writestring
			call writedec
			add var,1
			call crlf
		L2:
			add esi,4
			inc bl
			loop l1
		main ENDP
```

```
        exit


Get_frequencies PROC targets: PTR dword, freqT: PTR dword
mov ebp, esp
mov esi, targets
mov ecx, esi
mov esi, freqT

L1:
        mov ebx, [ecx]
        movzx eax, bl
        cmp eax,0
        je done
        mov edx, 4
        imul edx
        mov edx, esi
        add edx, eax
        mov eax, [edx]
        inc eax
        mov [edx], eax
        inc ecx
        jmp L1

done:
        ret
        Get_frequencies ENDP
END main
```