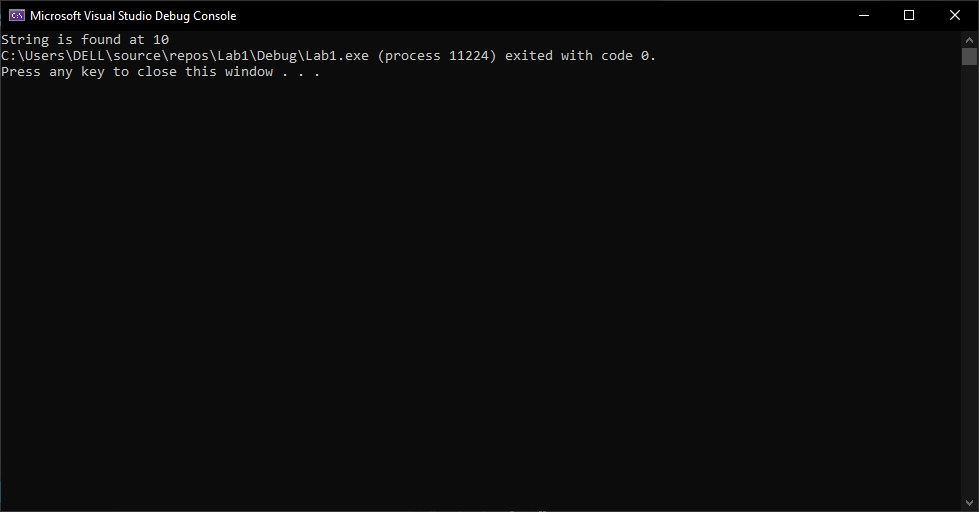
**Coal Lab 10**

**K226007**

**Task 1:**



**include irvine32.inc**

**.data**

**str1 byte "127&j~3@^&\*@\*@45^",0**

**p1 byte "String is found at ",0**

**p2 byte "String Not found",0**

**.code**

**main proc**

**call Scan\_string**

**exit**

**main endp**

**Scan\_string proc**

**mov edi, offset str1**

**mov ecx, lengthof str1**

**mov eax, 0**

**mov al, '@'**

**repne scasb**

**je found**

**jne not\_found**

**found:**

**mov edx, offset p1**

**mov eax, ecx**

**call WriteString**

**call WriteDec**

**jmp next**

**not\_found:**

**mov edx, offset p2**

**call writestring**

**next:**

**ret**

**Scan\_string endp**

**end main**  
  
  
**task 2:**

**include irvine32.inc**

**.data**

**str1 byte "127&j~3@^&\*@\*@45^",0**

**p1 byte "String is found at ",0**

**p2 byte "String Not found",0**

**proto Scan\_string, str1: byte, target: byte**

**.code**

**main proc**

**mov eax, 0**

**mov al, '@'**

**invoke Scan\_string, addr str1, al**

**exit**

**main endp**

**Scan\_string proc str1: byte, target: byte**

**mov eax, 0**

**mov al, target**

**mov edi, offset str1**

**mov ecx, lengthof str1**

**mov eax, 0**

**repne scasb**

**je found**

**jne not\_found**

**found:**

**mov edx, offset p1**

**mov eax, ecx**

**call WriteString**

**call WriteDec**

**jmp next**

**not\_found:**

**mov edx, offset p2**

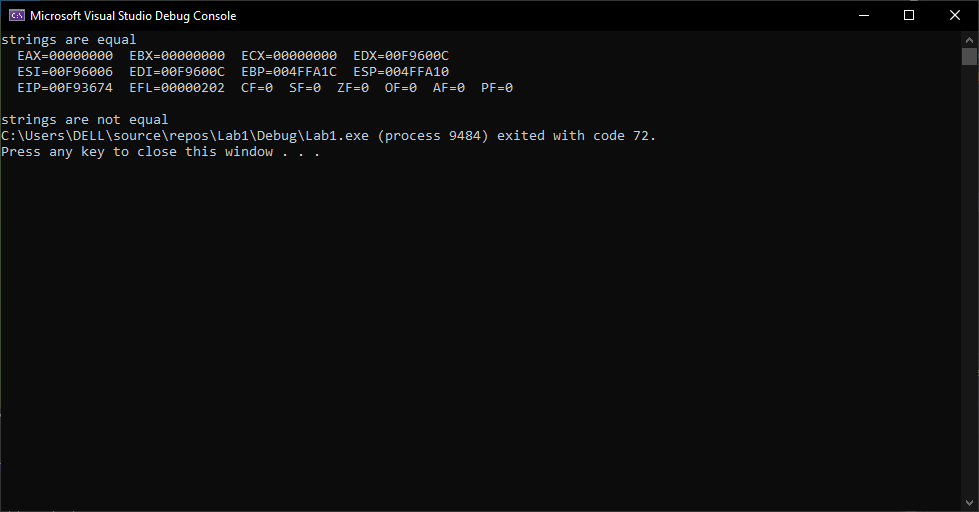
**call writestring**

**next:**

**ret**

**Scan\_string endp**

**end main**  
  
  
**note: invoke and proto not working on irvine32.inc**  
  
  
**task 3:**



**TITLE ReverseArray.asm**

**INCLUDE Irvine32.inc**

**.data**

**str\_1 byte "Hello",0**

**str\_2 byte "Hello",0**

**prompt byte "strings are equal",0**

**prompt\_2 byte "strings are not equal",0**

**.code**

**main PROC**

**mov esi, offset str\_1**

**mov edi, offset str\_2**

**call is\_compare**

**call dumpregs**

**main ENDP**

**is\_compare proc**

**mov ecx, lengthof str\_1**

**mov eax, 0**

**mov ebx, 0**

**l1:**

**mov al, [esi]**

**mov bl, [edi]**

**cld**

**cmp al, bl**

**jne str\_diff**

**inc esi**

**inc edi**

**loop l1**

**mov edx, offset prompt**

**call WriteString**

**jmp done**

**str\_diff:**

**mov edx, offset prompt\_2**

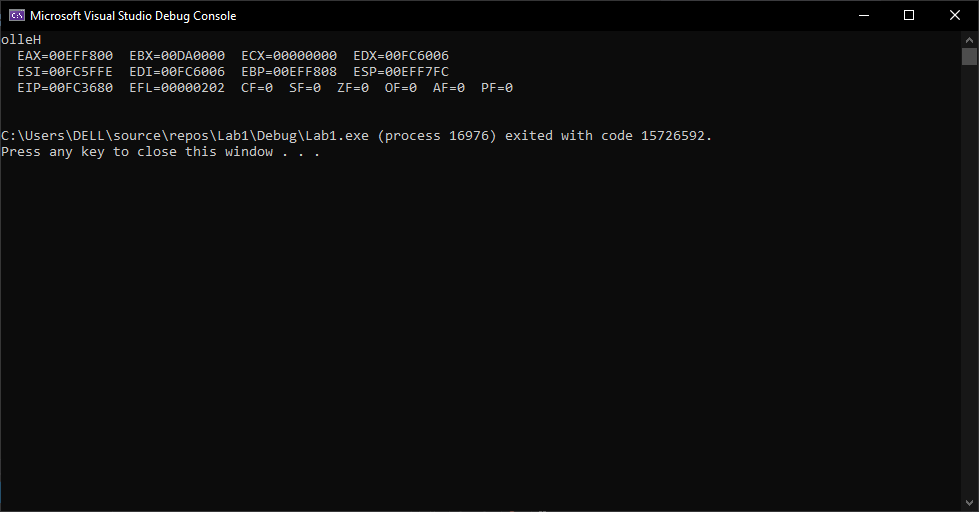
**call WriteString**

**done: ret**

**is\_compare endp**

**END main**

**Task 4:**



**TITLE ReverseArray.asm**

**INCLUDE Irvine32.inc**

**.data**

**str1 byte "Hello",0**

**str\_rev byte 100 dup(0)**

**.code**

**main PROC**

**mov esi, offset str1**

**mov edi, offset str\_rev**

**call string\_reverse**

**call dumpregs**

**main ENDP**

**string\_reverse proc**

**mov ecx, lengthof str1**

**l1:**

**inc esi**

**loop l1**

**dec esi**

**dec esi**

**mov ecx, lengthof str1**

**reverse:**

**mov al, [esi]**

**mov [edi], al**

**inc edi**

**dec esi**

**loop reverse**

**mov edi, offset str\_rev**

**mov edx, edi**

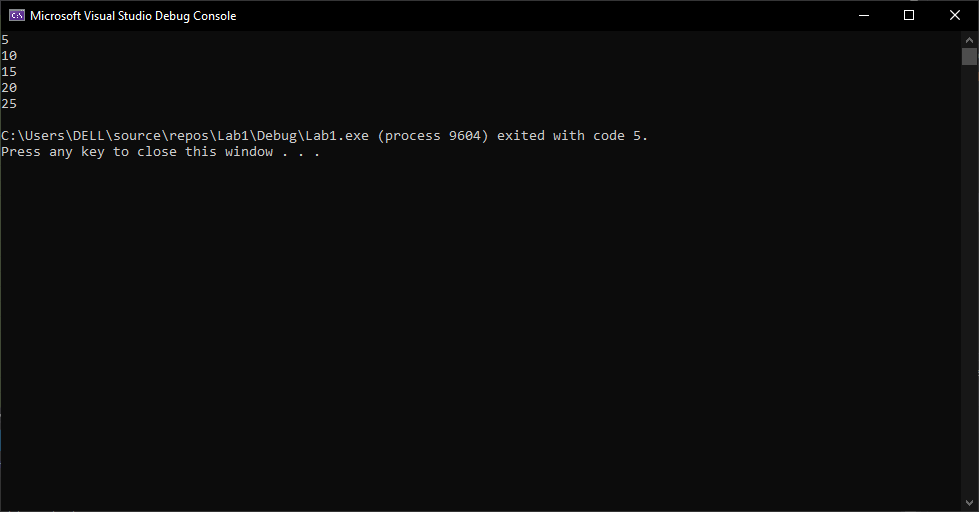
**call writeString**

**ret**

**string\_reverse endp**

**END main**

**Task 5:**



**include irvine32.inc**

**.data**

**array dword 1,2,3,4,5**

**mult byte 5**

**.code**

**main PROC**

**call load\_array**

**mov esi, offset array**

**mov ecx, lengthof array**

**l2:**

**mov eax, [esi]**

**call writedec**

**call crlf**

**add esi, 4**

**loop l2**

**main ENDP**

**load\_array proc**

**mov esi, offset array**

**mov ecx, lengthof array**

**movzx eax, mult**

**l1:**

**mov ebx, [esi]**

**imul ebx, eax**

**mov [esi], ebx**

**add esi, 4**

**loop l1**

**ret**

**load\_array endp**

**END main**

**Task 6:**

**include irvine32.inc**

**.data**

**target BYTE "AAEBDCFBBC", 0**

**freqTable DWORD 256 DUP (0)**

**.code**

**main PROC**

**INVOKE GetFrequency, ADDR target, ADDR freqTable**

**mov esi, OFFSET freqTable**

**mov ecx, 256**

**display\_loop:**

**movzx eax, BYTE PTR [esi]**

**cmp eax, 0**

**je done\_display**

**mov edx, esi**

**call WriteChar**

**mov eax, DWORD PTR [esi]**

**call WriteInt**

**call Crlf**

**add esi, 4**

**loop display\_loop**

**done\_display:**

**INVOKE ExitProcess, 0**

**main ENDP**

**GetFrequency PROC USES esi, target:PTR BYTE, freqTable:PTR DWORD**

**mov esi, freqTable**

**mov ecx, 256**

**rep stosd**

**mov esi, target**

**count\_loop:**

**movzx eax, BYTE PTR [esi]**

**cmp eax, 0**

**je end\_count**

**movzx edx, eax**

**mov eax, DWORD PTR [freqTable + edx \* 4]**

**inc eax**

**mov DWORD PTR [freqTable + edx \* 4], eax**

**inc esi**

**jmp count\_loop**

**end\_count:**

**ret**

**GetFrequency ENDP**

**END main**

**note: invoke and proto are not working in irvine32.inc**