COAL LAB 07 - 29 Mar 2025

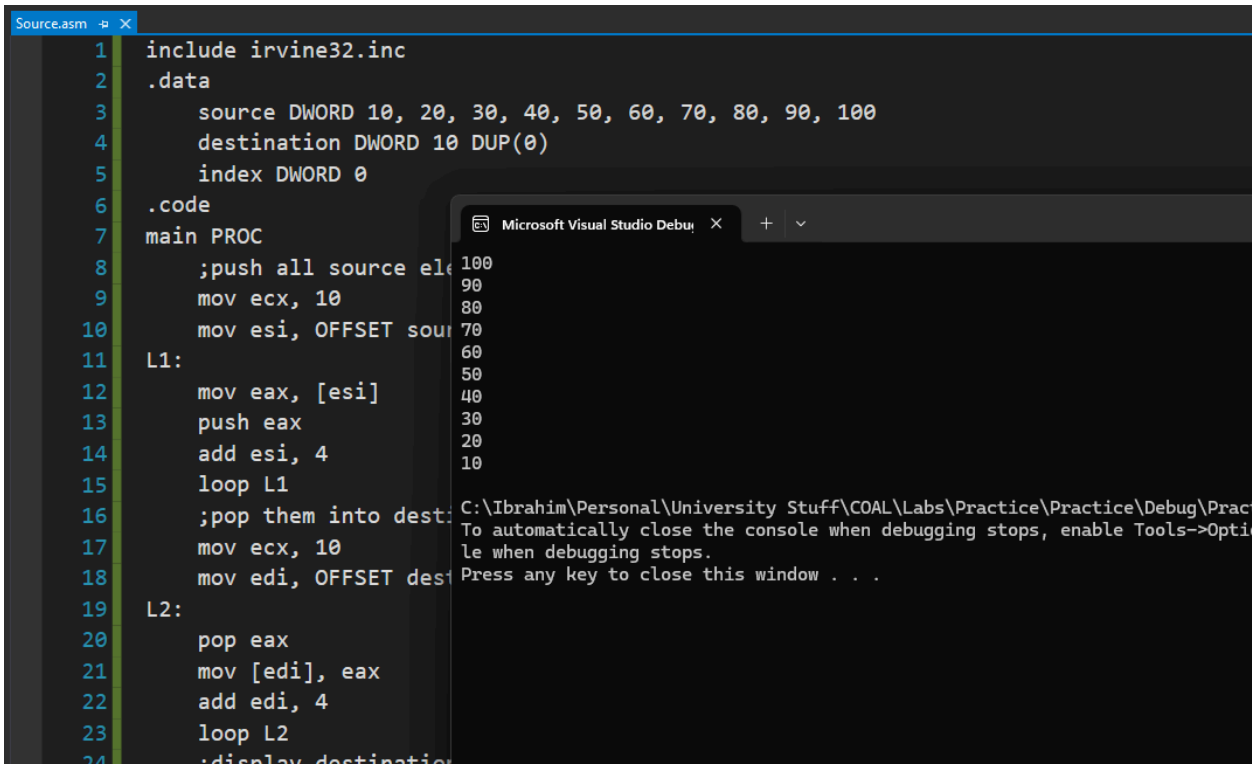
Student Name: Ibrahim Johar Farooqi

Student ID: 23K-0074

Task 1:

```
include irvine32.inc
.data
   source DWORD 10, 20, 30, 40, 50, 60, 70, 80, 90, 100
   destination DWORD 10 DUP(0)
   index DWORD 0
.code
main PROC
   ;push all source elements to stack
   mov ecx, 10
   mov esi, OFFSET source
L1:
   mov eax, [esi]
   push eax
   add esi, 4
   loop L1
   ;pop them into destination array
   mov ecx, 10
   mov edi, OFFSET destination
L2:
   pop eax
   mov [edi], eax
   add edi, 4
   loop L2
   ;display destination array
   mov ecx, 10
   mov esi, OFFSET destination
L3:
   mov eax, [esi]
   call WriteDec
   call Crlf
   add esi, 4
```

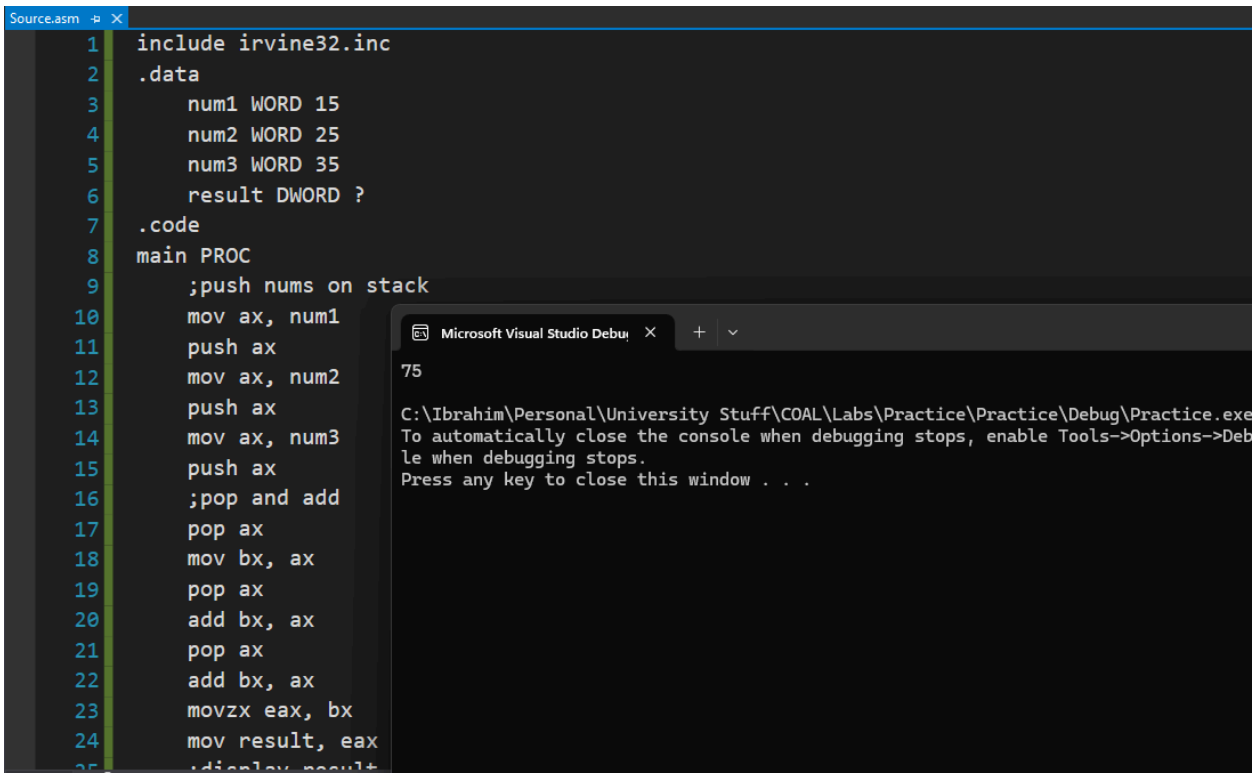
```
    loop L3
    exit
main ENDP
END main
```



Task 2:

```
include irvine32.inc
.data
    num1 WORD 15
    num2 WORD 25
    num3 WORD 35
    result DWORD ?
.code
main PROC
    ;push nums on stack
    mov ax, num1
    push ax
    mov ax, num2
    push ax
    mov ax, num3
    push ax
    ;pop and add
```

```
    pop ax
    mov bx, ax
    pop ax
    add bx, ax
    pop ax
    add bx, ax
    movzx eax, bx
    mov result, eax
    ;display result
    mov eax, result
    call WriteDec
    call Crlf
    exit
main ENDP
END main
```



Task 3:

```
include irvine32.inc
.data
    array1 DWORD 1, 2, 3, 4, 5
    array2 DWORD 6, 7, 8, 9, 10
    sum1 DWORD ?
```

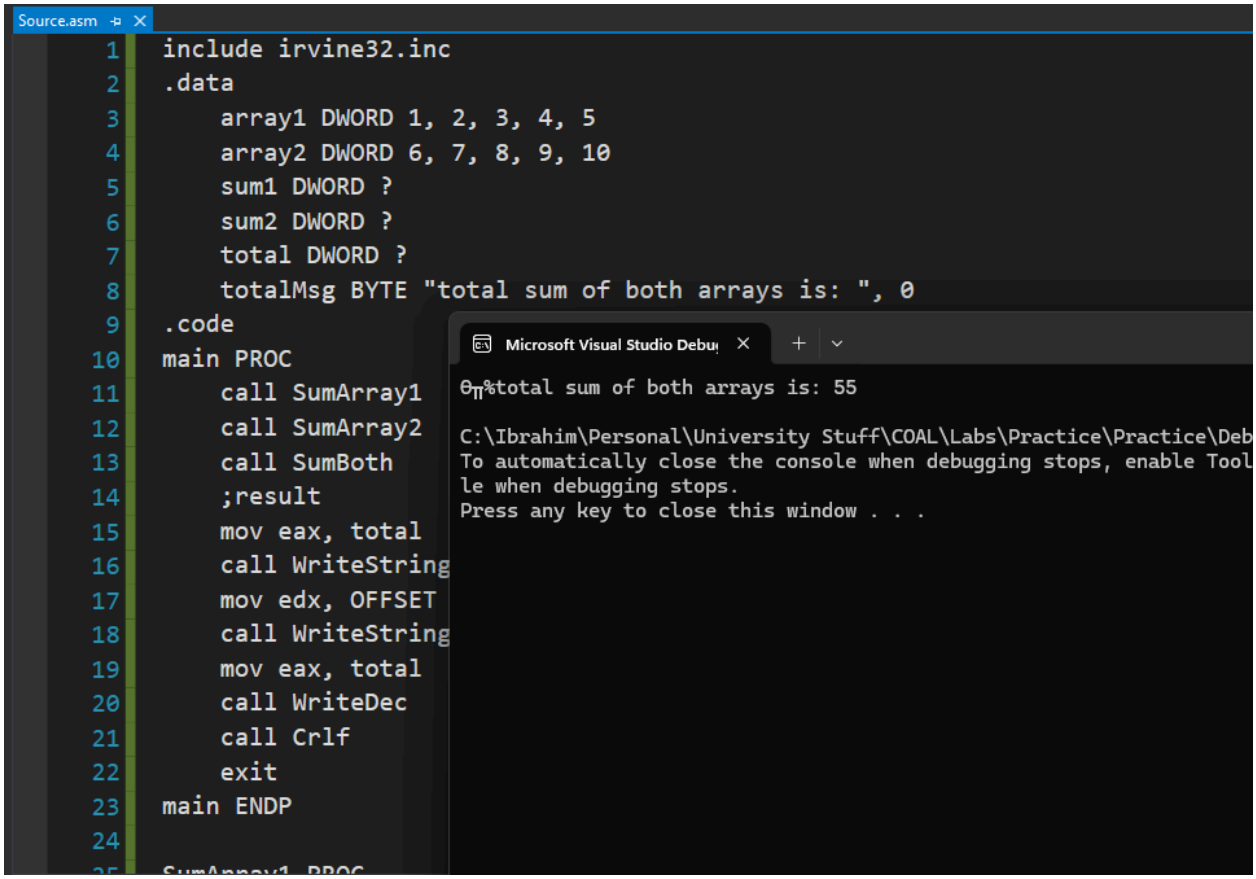
```
    sum2 DWORD ?
    total DWORD ?
    totalMsg BYTE "total sum of both arrays is: ", 0
.code
main PROC
    call SumArray1
    call SumArray2
    call SumBoth
    ;result
    mov eax, total
    call WriteString
    mov edx, OFFSET totalMsg
    call WriteString
    mov eax, total
    call WriteDec
    call Crlf
    exit
main ENDP

SumArray1 PROC
    mov ecx, 5
    mov esi, OFFSET array1
    xor eax, eax     ;clearing eax to accumulate sum
L1:
    add eax, [esi]
    add esi, 4
    loop L1
    mov sum1, eax
    ret
SumArray1 ENDP

SumArray2 PROC
    mov ecx, 5
    mov esi, OFFSET array2
    xor eax, eax
L2:
```

```
    add eax, [esi]
    add esi, 4
    loop L2
    mov sum2, eax
    ret
SumArray2 ENDP

SumBoth PROC
    mov eax, sum1
    add eax, sum2
    mov total, eax
    ret
SumBoth ENDP
END main
```



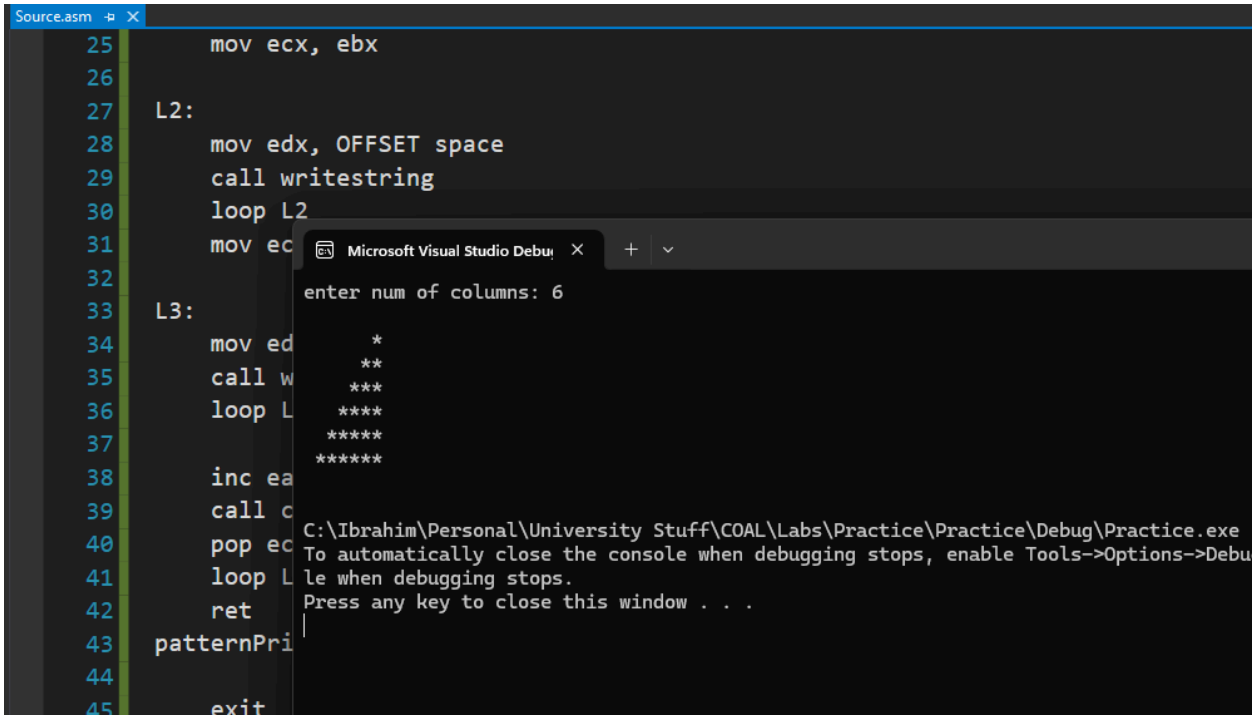
Task 4:

```
include irvine32.inc
.data
	star BYTE "*", 0
	space BYTE " ", 0
	input BYTE "enter num of columns: ", 0
	col DWORD ?
.code
main PROC
	mov edx, OFFSET input
	call writestring
	call readdec
	mov col, eax
	call crlf
	call patternPrint
	call crlf
	exit
main ENDP

patternPrint PROC
	mov ecx, col
	mov eax, 1
L1:
	push ecx
	mov ebx, ecx
	mov ecx, ebx
L2:
	mov edx, OFFSET space
	call writestring
	loop L2
	mov ecx, eax
L3:
	mov edx, OFFSET star
	call writestring
	loop L3
	inc eax
```

```
        call crlf
        pop ecx
        loop L1
        ret
patternPrint ENDP

        exit
END main
```



Task 5:

```
include irvine32.inc
.data
    n DWORD ?
    msg byte "enter a num: ", 0
    msg2 byte "sum of numbers from 1 till your num: ", 0
.code
main PROC
    mov edx, offset msg
    call writestring
    call readdec
    call sumN
    mov edx, offset msg2
    call writestring
```

```
    call writedec
    exit
main ENDP

sumN PROC
    push ecx
    push ebx
    mov ecx, eax
    mov eax, 0
    mov ebx, 1
L1:
    add eax, ebx
    inc ebx
    loop L1
    pop ebx
    pop ecx
    ret
sumN ENDP
END main
```