

COAL-LAB 8

TASK 1:

```
INCLUDE Irvine32.inc
.data
msg1 byte "Integers Found Equal to number ",0
msg2 byte "No Integer is Equal!",0
sign byte "=",0
integers dword ?
larger dword ?
x dword ?
y dword ?
z dword ?
w dword ?
```

```
.code
main PROC
mov w,0
mov esi, offset integers
mov z,esi
mov ecx,4
```

```
input:
call readint
mov [esi],eax
add esi,4
loop input
mov esi,z
```

```
mov ecx,4
mov edi, offset integers
```

```
call crlf
```

```
mov z,4
compare:
cmp w, 4
je fin
mov eax,[edi]
```

```
mov ecx,z
cmp z,0
je fin
```

```
compare2:
add esi,4
mov ebx,[esi]
mov x,eax
cmp eax, ebx
je cond
call crlf
mov edx,offset msg2
call writestring
```

```
loop compare2
```

```
add edi,4
sub z,1
```

```
loop compare
```

```
cond:
add edi,4
add w,1
cmp w,1
jg equal
jmp compare2
loop cond
```

```
equal:
call crlf
mov eax,w
call writeint
mov edx,offset msg1
call writestring
mov eax,x
call writeint
add esi,4
jmp compare
cmp w, 4
je fin
loop equal
```

```
fin:
```

exit
loop fin

main ENDP
END main

```
my code.asm  X
1  INCLUDE Irvine32.inc
2  .data
3  msg1 byte "Integers Found Equal to number ",0
4  msg2 byte "No Integer is Equal!",0
5  sign byte " = ",0
6  integers dword ?
7  larger dword ?
8  x dword ?
9  y dword ?
10 z dword ?
11 w dword ?
12
13 .code
14 main PROC
15 mov w,0
16 mov esi, offset integers
17 mov z,esi
18 mov ecx,4
19
20 input:
21 call readint
22 mov [esi],eax
23 add esi,4
24 loop input
25 mov esi,z
26
27 mov ecx,4
28 mov edi, offset integers
29
30
31
32 call crlf
33
34 mov z,4
35 compare:
36 cmp w, 4
37 je fin
38 mov eax,[edi]
39
```

```
Microsoft Visual Studio Debug Console
1
2
1
3
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
+2Integers Found Equal to number +1
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
No Integer is Equal!
C:\Users\BLACKLIST\source\repos\Project
d with code 0.
Press any key to close this window . .
```

TASK 2:

```
INCLUDE Irvine32.inc
.data
msg1 byte "First Non Zero value Found = ",0

intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0

.code
main PROC

mov ecx, lengthof intArr
mov esi,offset intArr

nonzero:
mov eax,[esi]
mov edx,0
mov ebx,10
div ebx

cmp edx, 0
jne fin

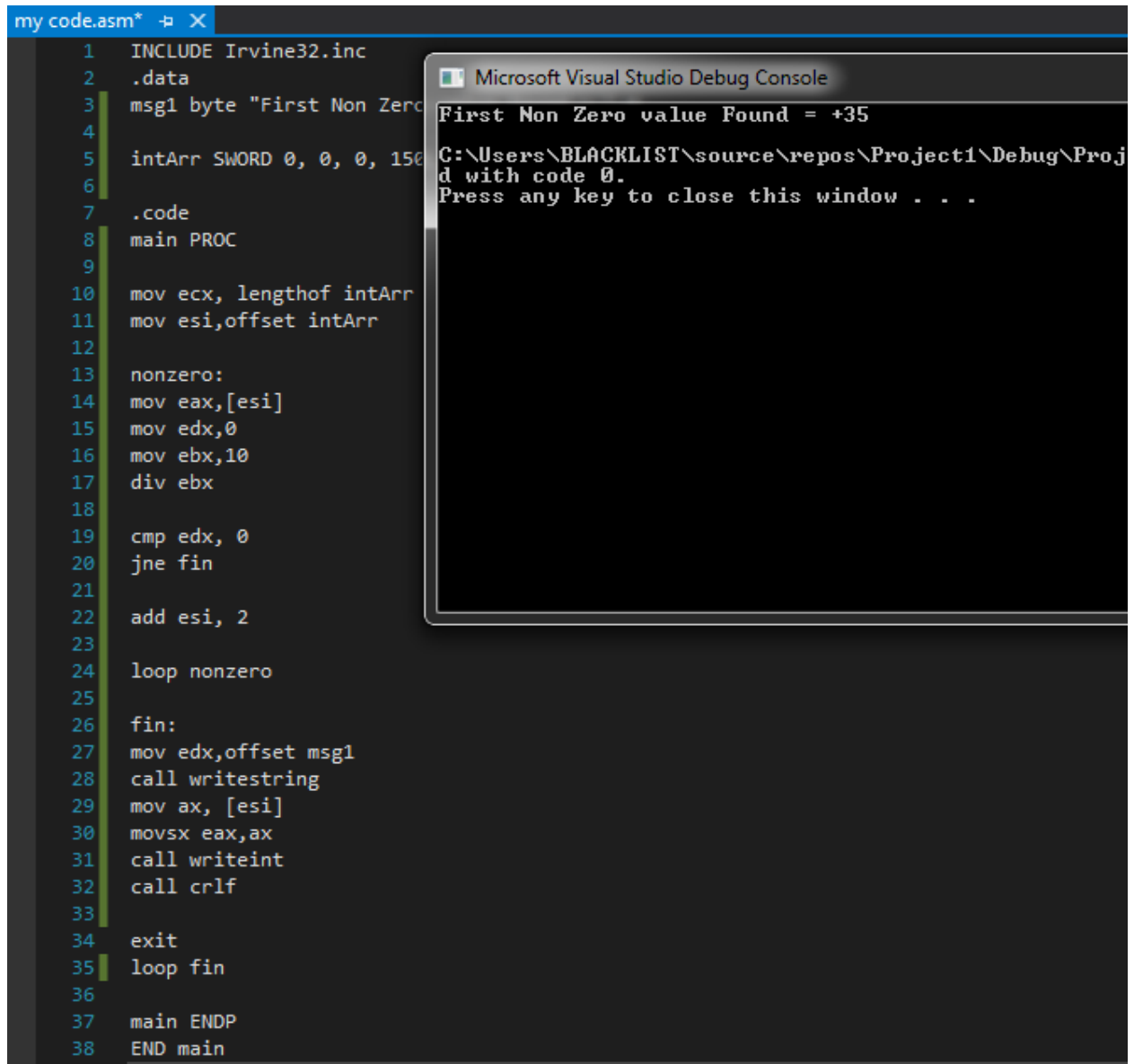
add esi, 2

loop nonzero

fin:
mov edx,offset msg1
call writestring
mov ax, [esi]
movsx eax,ax
call writeint
call crlf

exit
loop fin

main ENDP
END main
```



```
my code.asm*  -  X
1  INCLUDE Irvine32.inc
2  .data
3  msg1 byte "First Non Zero"
4
5  intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0
6
7  .code
8  main PROC
9
10 mov ecx, lengthof intArr
11 mov esi, offset intArr
12
13 nonzero:
14 mov eax, [esi]
15 mov edx, 0
16 mov ebx, 10
17 div ebx
18
19 cmp edx, 0
20 jne fin
21
22 add esi, 2
23
24 loop nonzero
25
26 fin:
27 mov edx, offset msg1
28 call writestring
29 mov ax, [esi]
30 movsx eax, ax
31 call writeint
32 call crlf
33
34 exit
35 loop fin
36
37 main ENDP
38 END main
```

Microsoft Visual Studio Debug Console

First Non Zero value Found = +35

C:\Users\BLACKLIST\source\repos\Project1\Debug\Project1.exe: terminated with code 0.

Press any key to close this window . . .

TASK 3:

INCLUDE Irvine32.inc

.data

intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0

var dword ?

x dword ?

y dword ?

.code

main PROC

```
mov var,5
mov ecx,lengthof intArr
mov ebx,ecx
mov y, -1
l1:
cmp y,ecx
jge fin
add y,1
```

```
call crlf
add var,1
mov edx,var
cmp ecx,0
je fin
```

```
cmp var,ecx
jl l2
```

```
l2:
cmp ecx,edx
jge l3
mov x,1
call else1
```

```
l3:
mov x,0
mov eax,x
call writedec
call crlf
jmp l1
loop l3
```

```
loop l1
```

```
else1:
```

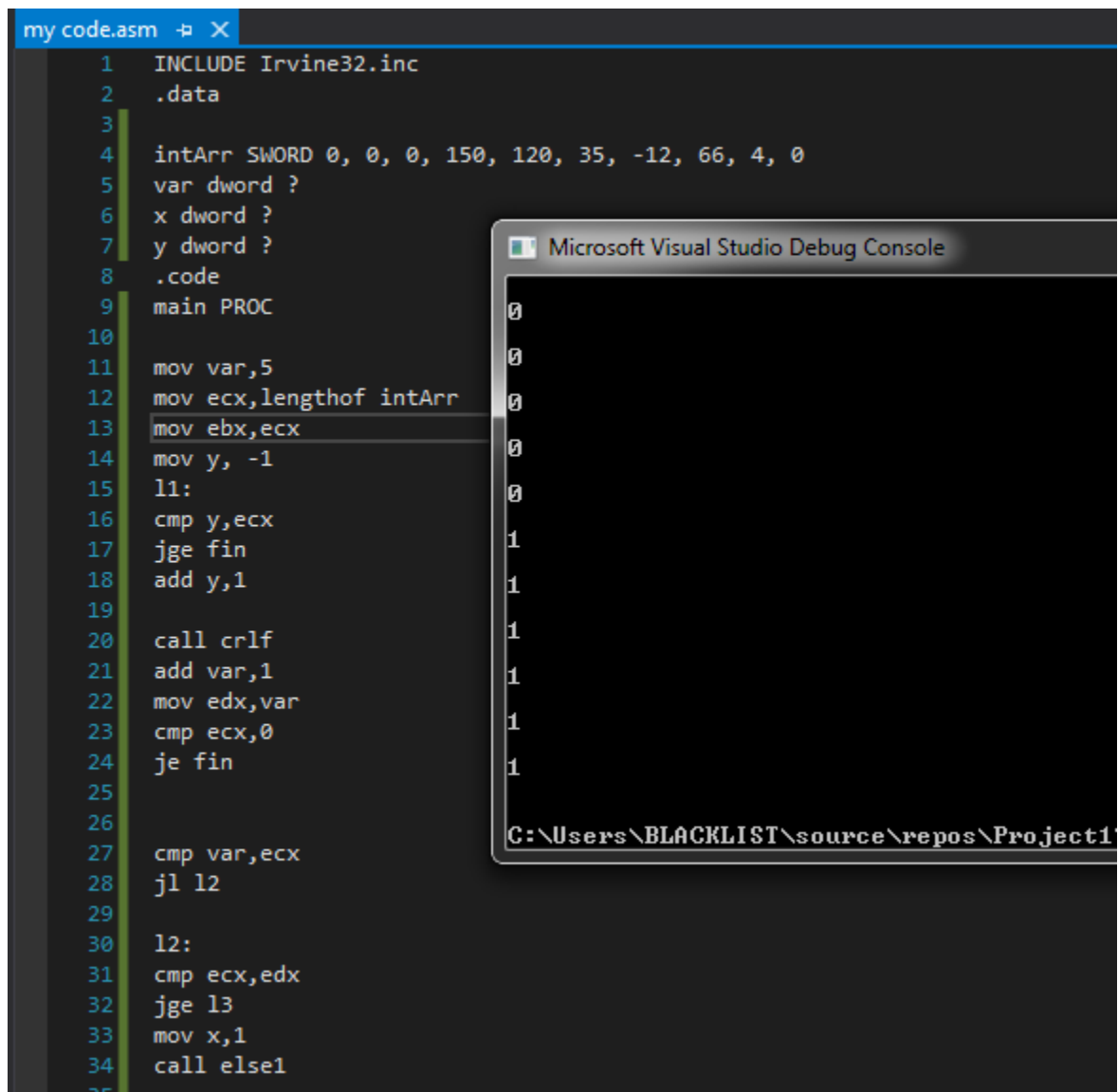
```
mov eax,x
call writedec
call crlf
```

```
jmp l1
loop else1
```

```
fin:
```

call crlf
exit
loop fin

main ENDP
END main



The screenshot shows a Visual Studio code editor with a file named 'my code.asm'. The code is written in assembly language and includes Irvine32.inc. It defines a data segment with an array 'intArr' and a code segment with a 'main' procedure. The 'main' procedure initializes a counter 'var' to 5, sets up a loop, and calls 'crlf' to print a newline character for each element in the array. A debug console window is open, showing the output of the program, which consists of ten zeros, indicating that the array elements are not being printed correctly. The path to the project is visible at the bottom of the console window.

```
1  INCLUDE Irvine32.inc
2  .data
3
4  intArr SWORD 0, 0, 0, 150, 120, 35, -12, 66, 4, 0
5  var dword ?
6  x dword ?
7  y dword ?
8  .code
9  main PROC
10
11  mov var,5
12  mov ecx,lengthof intArr
13  mov ebx,ecx
14  mov y, -1
15  l1:
16  cmp y,ecx
17  jge fin
18  add y,1
19
20  call crlf
21  add var,1
22  mov edx,var
23  cmp ecx,0
24  je fin
25
26
27  cmp var,ecx
28  jl l2
29
30  l2:
31  cmp ecx,edx
32  jge l3
33  mov x,1
34  call else1
35
```

Microsoft Visual Studio Debug Console

0
0
0
0
0
0
1
1
1
1
1
1
1
1
1
1
C:\Users\BLACKLIST\source\repos\Project1

TASK 4:

INCLUDE Irvine32.inc

```
.data
msg1 byte "-HELLO-",0
msg2 byte "-WORLD-",0
var dword ?
.code
main PROC
mov var,-1
```

```
l1:
add var,1 ; Here, var=0 on first iteration
cmp var , 10
jg fin
```

```
mov eax,var
call writedec
```

```
cmp var ,5
jl hello
```

```
cmp var ,5
jge world
```

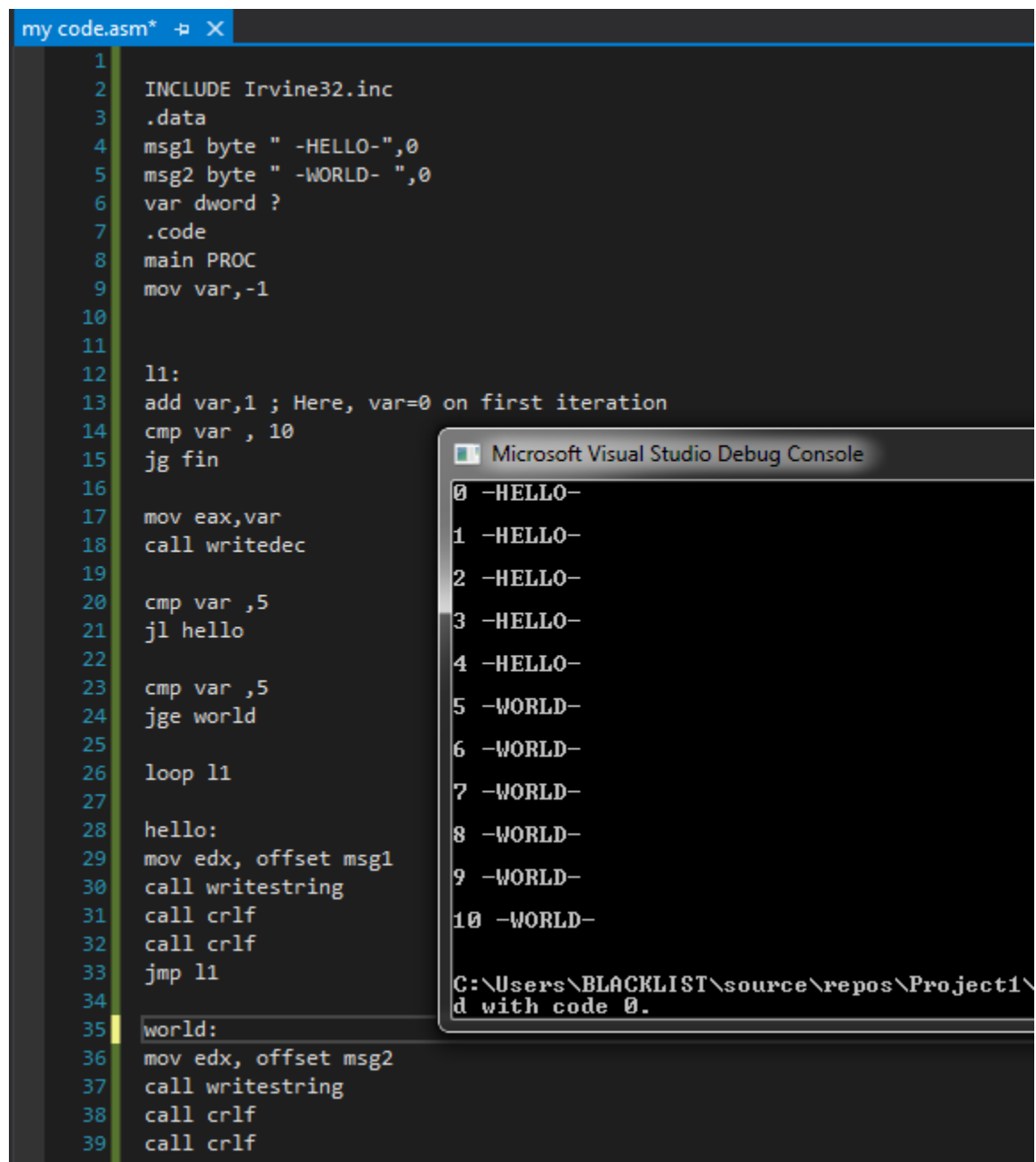
```
loop l1
```

```
hello:
mov edx, offset msg1
call writestring
call crlf
call crlf
jmp l1
```

```
world:
mov edx, offset msg2
call writestring
call crlf
call crlf
jmp l1
```

```
fin:
exit
loop fin
```

```
main ENDP
END main
```

The screenshot shows a Visual Studio IDE with a file named 'my code.asm' open. The assembly code is as follows:

```
1
2 INCLUDE Irvine32.inc
3 .data
4 msg1 byte " -HELLO-",0
5 msg2 byte " -WORLD- ",0
6 var dword ?
7 .code
8 main PROC
9 mov var,-1
10
11
12 l1:
13 add var,1 ; Here, var=0 on first iteration
14 cmp var , 10
15 jg fin
16
17 mov eax,var
18 call writedec
19
20 cmp var ,5
21 jl hello
22
23 cmp var ,5
24 jge world
25
26 loop l1
27
28 hello:
29 mov edx, offset msg1
30 call writestring
31 call crlf
32 call crlf
33 jmp l1
34
35 world:
36 mov edx, offset msg2
37 call writestring
38 call crlf
39 call crlf
```

The 'Microsoft Visual Studio Debug Console' window shows the output of the program:

```
0 -HELLO-
1 -HELLO-
2 -HELLO-
3 -HELLO-
4 -HELLO-
5 -WORLD-
6 -WORLD-
7 -WORLD-
8 -WORLD-
9 -WORLD-
10 -WORLD-
```

The console also shows the file path: C:\Users\BLACKLIST\source\repos\Project1\ and the message: d with code 0.

TASK 5:

INCLUDE Irvine32.inc

.data

msg1 byte "Please enter an Integer = ",0
msg2 byte "Integer match Found !",0
msg3 byte "No Integer match Found !",0
arr WORD 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20

x dword ?
y dword ?

```
.code  
main PROC  
mov edx ,offset msg1  
call writestring  
call readint  
mov x,eax
```

```
mov ecx, lengthof arr  
mov esi, offset arr  
mov y,esi
```

```
call crlf  
display:  
mov ax,[esi]  
movzx eax,ax
```

```
call writedec  
call crlf  
add esi,2  
loop display
```

```
mov eax,x  
mov esi,y  
mov ecx, lengthof arr
```

```
check:  
mov bx,[esi]  
movzx ebx,bx
```

```
cmp ax,bx  
je found  
add esi,2
```

```
loop check
```

```
jmp notfound
```

```
found:
```

```
call crlf
mov edx , offset msg2
call writestring
call crlf
exit
loop found
```

```
notfound:
call crlf
mov edx , offset msg3
call writestring
call crlf
exit
loop notfound
```

```
main ENDP
END main
```

my code.asm

```
1
2   INCLUDE Irvine32.inc
3
4   .data
5   msg1 byte "Please enter an Integer = ",0
6   msg2 byte "Integer match Found !",0
7   msg3 byte "No Integer match Found !",0
8   arr WORD 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20
9
10  x dword ?
11  y dword ?
12
13  .code
14  main PROC
15      mov edx,offset msg1
16      call writestring
17      call readint
18      mov x,eax
19
20      mov ecx, lengthof arr
21      mov esi, offset arr
22      mov y,esi
23
24      call crlf
25  display:
26      mov ax,[esi]
27      movzx eax,ax
28
29      call writedec
30      call crlf
31      add esi,2
32      loop display
33
34      mov eax,x
35      mov esi,y
36      mov ecx, lengthof arr
37
```

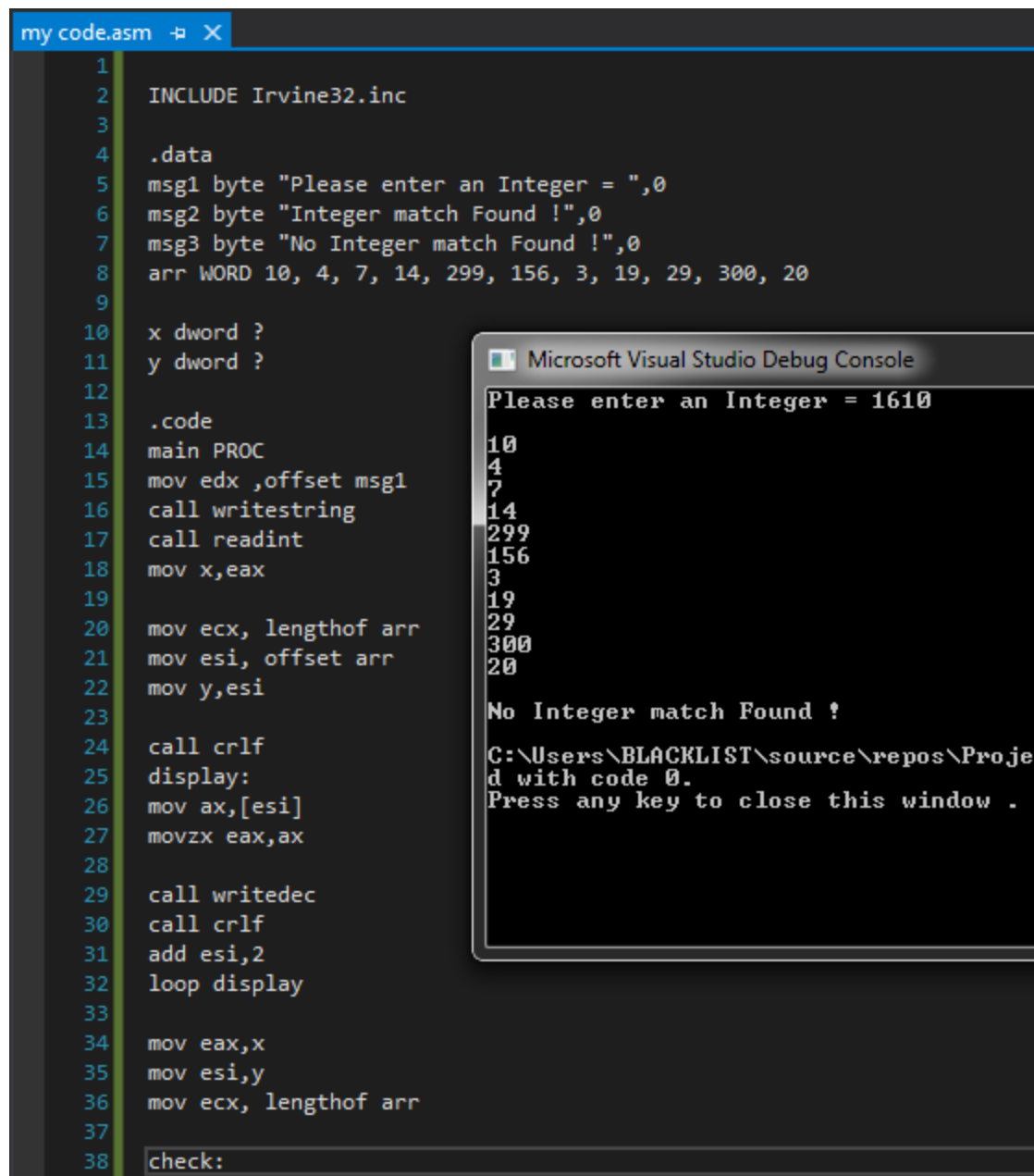
Microsoft Visual Studio Debug Console

Please enter an Integer = 299

10
4
7
14
299
156
3
19
29
300
20

Integer match Found !

C:\Users\BLACKLIST\source\repos\Pro
d with code 0.
Press any key to close this window



The screenshot shows a Visual Studio IDE with a file named 'my code.asm' open. The assembly code is as follows:

```
1
2 INCLUDE Irvine32.inc
3
4 .data
5 msg1 byte "Please enter an Integer = ",0
6 msg2 byte "Integer match Found !",0
7 msg3 byte "No Integer match Found !",0
8 arr WORD 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20
9
10 x dword ?
11 y dword ?
12
13 .code
14 main PROC
15 mov edx,offset msg1
16 call writestring
17 call readint
18 mov x,eax
19
20 mov ecx,lengthof arr
21 mov esi,offset arr
22 mov y,esi
23
24 call crlf
25 display:
26 mov ax,[esi]
27 movzx eax,ax
28
29 call writedec
30 call crlf
31 add esi,2
32 loop display
33
34 mov eax,x
35 mov esi,y
36 mov ecx,lengthof arr
37
38 check:
```

The 'Microsoft Visual Studio Debug Console' window shows the program's output:

```
Please enter an Integer = 1610
10
4
7
14
299
156
3
19
29
300
20

No Integer match Found !

C:\Users\BLACKLIST\source\repos\Project with code 0.
Press any key to close this window .
```

TASK 6:

INCLUDE Irvine32.inc

.data

arr WORD 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20

i DWORD 1

temp DWORD ?

```

.code
main PROC
mov ecx, lengthof arr
l1:
mov temp, ecx
mov eax, 0
mov ebx, 0
sub ecx, i
mov esi, 0
mov edi, 1

l2:
mov ax, arr[esi*TYPE arr]
mov bx, arr[edi*TYPE arr]
cmp ax, bx
JG exchange
JLE cont

exchange:
mov ax, arr[esi*TYPE arr]
mov bx, arr[edi*TYPE arr]
xchg ax, arr[edi*TYPE arr]
xchg bx, arr[esi*TYPE arr]

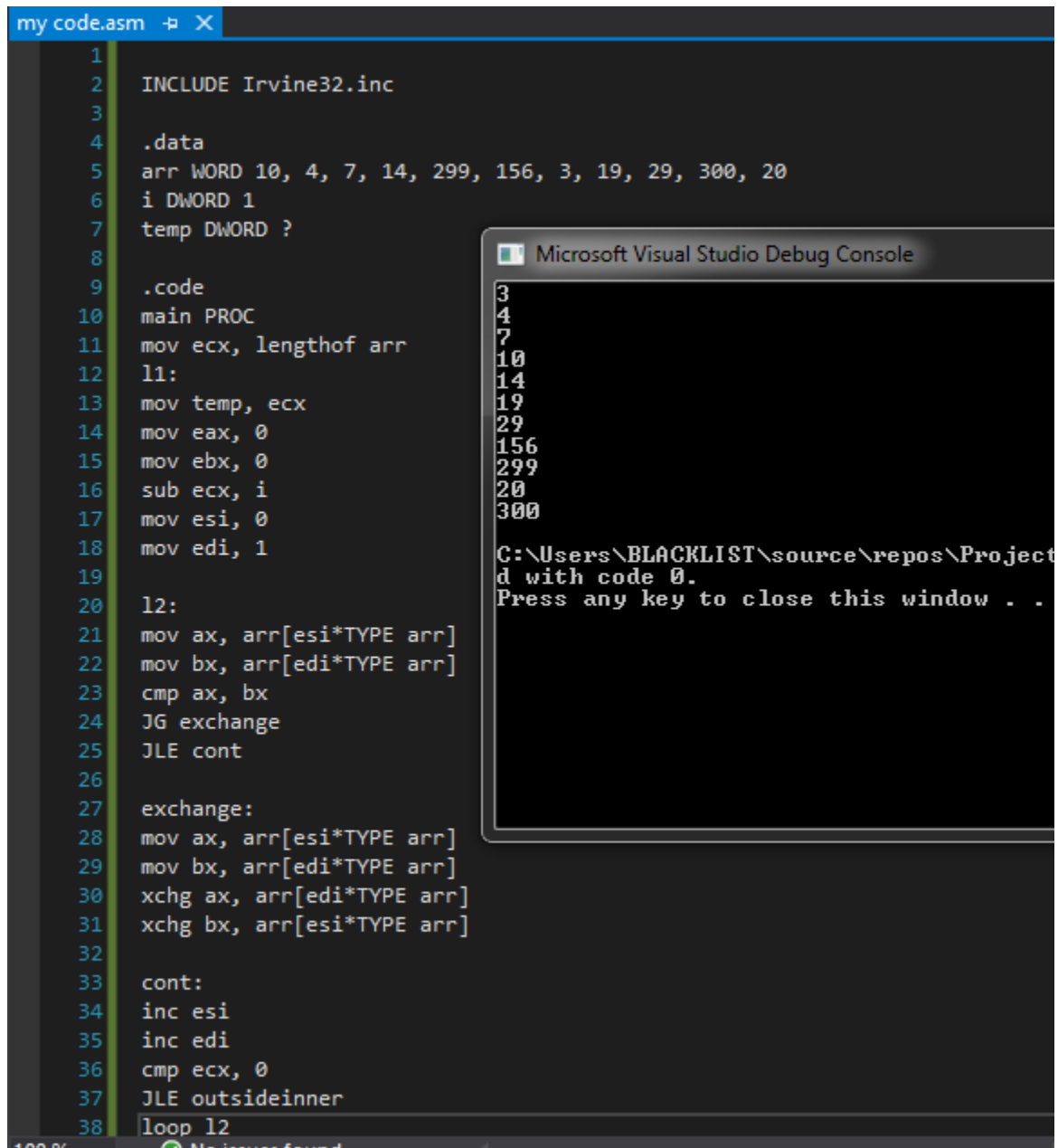
cont:
inc esi
inc edi
cmp ecx, 0
JLE outsideinner
loop l2

outsideinner:
mov ecx, temp
inc i
loop l1
mov edi, 0
mov ecx, LENGTHOF arr
mov esi, 0

l3:
mov ax, arr[esi*TYPE arr]
inc esi
call writedec
call crlf
loop l3

```

```
exit
main ENDP
END main
```



The screenshot shows a Visual Studio IDE with a dark theme. The main window displays an assembly file named 'my code.asm'. The code includes Irvine32.inc, defines an array 'arr' of 10 words, and a loop that compares and exchanges elements. A 'Microsoft Visual Studio Debug Console' window is open in the foreground, displaying the contents of the array: 3, 4, 7, 10, 14, 19, 29, 156, 299, 20, 300. Below the array, it shows the file path 'C:\Users\BLACKLIST\source\repos\Project' and the message 'd with code 0. Press any key to close this window . . .'. The status bar at the bottom indicates '100%' zoom and 'No errors found'.

```
1
2 INCLUDE Irvine32.inc
3
4 .data
5 arr WORD 10, 4, 7, 14, 299, 156, 3, 19, 29, 300, 20
6 i DWORD 1
7 temp DWORD ?
8
9 .code
10 main PROC
11 mov ecx, lengthof arr
12 l1:
13 mov temp, ecx
14 mov eax, 0
15 mov ebx, 0
16 sub ecx, i
17 mov esi, 0
18 mov edi, 1
19
20 l2:
21 mov ax, arr[esi*TYPE arr]
22 mov bx, arr[edi*TYPE arr]
23 cmp ax, bx
24 JG exchange
25 JLE cont
26
27 exchange:
28 mov ax, arr[esi*TYPE arr]
29 mov bx, arr[edi*TYPE arr]
30 xchg ax, arr[edi*TYPE arr]
31 xchg bx, arr[esi*TYPE arr]
32
33 cont:
34 inc esi
35 inc edi
36 cmp ecx, 0
37 JLE outsideinner
38 loop l2
```

Microsoft Visual Studio Debug Console

3
4
7
10
14
19
29
156
299
20
300

C:\Users\BLACKLIST\source\repos\Project
d with code 0.
Press any key to close this window . . .

100% No errors found

TASK 7:

```
INCLUDE Irvine32.inc
.data
week byte "          ~Week Days With Numeric Order~",0
msg byte "Enter a Number between 1 - 7 : ",0
```

```

mon byte "          MONDAY",0
tue byte "          TUESDAY",0
wed byte "          WEDNESDAY",0
thurs byte "        THURSDAY",0
fri byte "          FRIDAY",0
sat byte "          SATURDAY",0
sun byte "          SUNDAY",0
invalid byte "Invalid entry! Please enter valid week-day number 1 - 7!",0
.code
main PROC
call crlf
mov edx, offset week
call writestring
call crlf

num:
call crlf
mov edx, offset msg
call writestring
call readdec

cmp eax,1
jl fin

cmp eax,1
je mo

cmp eax,2
je tu

cmp eax,3
je we

cmp eax,4
je th

cmp eax,5
je fr

cmp eax,6
je sa

cmp eax,7
je su

cmp eax,7

```


jge fin

loop num

mo:

mov edx, offset mon

call writestring

call crlf

jmp num

tu:

mov edx, offset tue

call writestring

call crlf

jmp num

we:

mov edx, offset wed

call writestring

call crlf

jmp num

th:

mov edx, offset thurs

call writestring

call crlf

jmp num

fr:

mov edx, offset fri

call writestring

call crlf

jmp num

sa:

mov edx, offset sat

call writestring

call crlf

jmp num

su:

mov edx, offset sun

call writestring

call crlf

jmp num

```

fin:
call crlf
mov edx, offset invalid
call writestring
call crlf
exit
loop fin

```

```

main ENDP
END main

```

The screenshot displays the Visual Studio IDE with an assembly file named 'my code.asm' open in the editor. The code defines a program that prompts the user to enter a week-day number (1-7) and prints the corresponding day name. It includes a loop to handle invalid input (number 8) by displaying an error message and looping back to the input prompt.

The assembly code is as follows:

```

1  INCLUDE Irvine32.inc
2  .data
3  week byte "    ~Week Days With Numeric Order~",0
4  msg byte "Enter a Number between 1 - 7 : ",0
5  mon byte "    MONDAY",0
6  tue byte "    TUESDAY",0
7  wed byte "    WEDNESDAY",0
8  thurs byte "    THURSDAY",0
9  fri byte "    FRIDAY",0
10 sat byte "    SATURDAY",0
11 sun byte "    SUNDAY",0
12 invalid byte " Invalid entry! Please enter valid week-day number 1 - 7 : ",0
13 .code
14 main PROC
15 call crlf
16 mov edx, offset week
17 call writestring
18 call crlf
19
20 num:
21 call crlf
22 mov edx, offset msg
23 call writestring
24 call readdec
25
26 cmp eax,1
27 jl fin
28
29 cmp eax,1
30 je mo
31
32 cmp eax,2
33 je tu
34
35 cmp eax,3
36 je we
37

```

The Microsoft Visual Studio Debug Console shows the program's execution output:

```

~Week Days With Numeric Order~
Enter a Number between 1 - 7 : 1
MONDAY
Enter a Number between 1 - 7 : 2
TUESDAY
Enter a Number between 1 - 7 : 3
WEDNESDAY
Enter a Number between 1 - 7 : 4
THURSDAY
Enter a Number between 1 - 7 : 5
FRIDAY
Enter a Number between 1 - 7 : 6
SATURDAY
Enter a Number between 1 - 7 : 7
SUNDAY
Enter a Number between 1 - 7 : 8
Invalid entry! Please enter valid week-day number 1 - 7 :
C:\Users\BLACKLIST\source\repos\Project1\Debug\Project1.exe
d with code 0.
Press any key to close this window . . .

```

TASK 8:

```
INCLUDE Irvine32.inc
```

```
.data
msg byte "Enter a character : ",0
msg2 byte "Character is an Alphabet!",0
msg3 byte "Character is not an Alphabet!",0
letter byte 65,0
```

```
.code
main PROC
call crlf
mov edx, offset msg
call writestring
call readchar
```

```
call writechar
call crlf
```

```
mov ecx,255
```

```
mov edx,offset letter
```

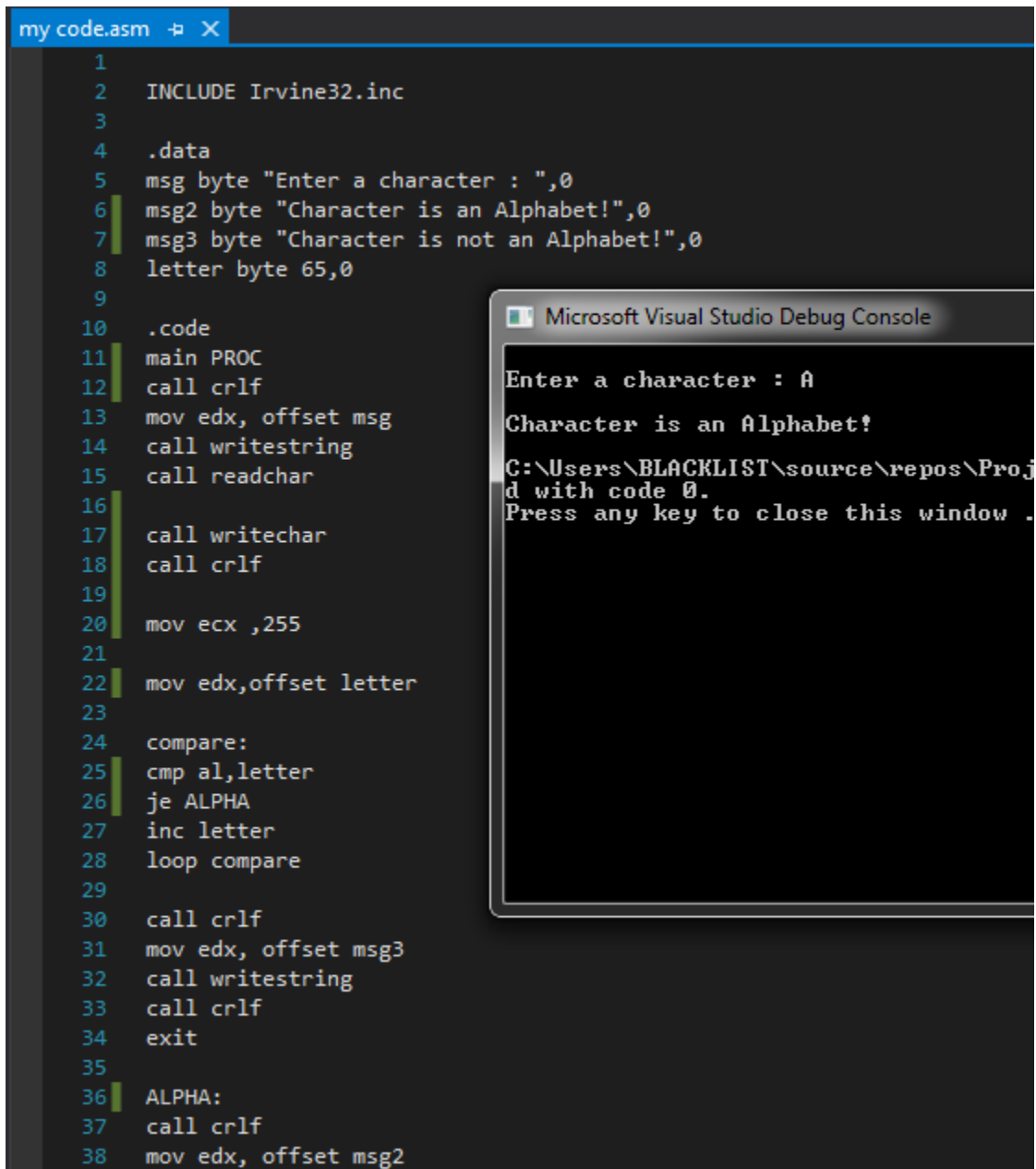
```
compare:
cmp al,letter
je ALPHA
inc letter
loop compare
```

```
call crlf
mov edx, offset msg3
call writestring
call crlf
exit
```

```
ALPHA:
call crlf
mov edx, offset msg2
call writestring
call crlf
exit
loop ALPHA
```

```
main ENDP
```

END main



The image shows a Visual Studio code editor window with a file named 'my code.asm'. The code is written in assembly language and includes Irvine32.inc. It defines a main PROC that prompts the user to enter a character, reads the character, and checks if it is an alphabet. If it is, it prints 'Character is an Alphabet!'. If not, it prints 'Character is not an Alphabet!'. The code also includes a loop to compare the character with a list of alphabet characters (A-Z) and a label ALPHA for the 'not an alphabet' case.

```
1
2  INCLUDE Irvine32.inc
3
4  .data
5  msg byte "Enter a character : ",0
6  msg2 byte "Character is an Alphabet!",0
7  msg3 byte "Character is not an Alphabet!",0
8  letter byte 65,0
9
10 .code
11 main PROC
12 call crlf
13 mov edx, offset msg
14 call writestring
15 call readchar
16
17 call writechar
18 call crlf
19
20 mov ecx ,255
21
22 mov edx,offset letter
23
24 compare:
25 cmp al,letter
26 je ALPHA
27 inc letter
28 loop compare
29
30 call crlf
31 mov edx, offset msg3
32 call writestring
33 call crlf
34 exit
35
36 ALPHA:
37 call crlf
38 mov edx, offset msg2
```

The 'Microsoft Visual Studio Debug Console' window is open, showing the output of the program. It displays the prompt 'Enter a character : A', followed by the message 'Character is an Alphabet!'. Below this, it shows the file path 'C:\Users\BLACKLIST\source\repos\Proj' and the message 'd with code 0. Press any key to close this window .'. The console window has a title bar that reads 'Microsoft Visual Studio Debug Console'.

```
my code.asm  + X
1
2  INCLUDE Irvine32.inc
3
4  .data
5  msg byte "Enter a character : ",0
6  msg2 byte "Character is an Alphabet!",0
7  msg3 byte "Character is not an Alphabet!",0
8  letter byte 65,0
9
10 .code
11 main PROC
12 call crlf
13 mov edx, offset msg
14 call writestring
15 call readchar
16
17 call writechar
18 call crlf
19
20 mov ecx,122
21
22 mov edx,offset letter
23
24 compare:
25 cmp al,letter
26 je ALPHA
27 inc letter
28 loop compare
29
30 call crlf
31 mov edx, offset msg3
32 call writestring
33 call crlf
34 exit
35
36 ALPHA:
37 call crlf
38 mov edx, offset msg2
39 call writestring
40 call crlf
41 exit
```

Microsoft Visual Studio Debug Console

Enter a character : 9

Character is not an Alphabet!

C:\Users\BLACKLIST\source\repos\Pro
d with code 0.

Press any key to close this window

NAME: MAQBOOL AHMED
ROLLNO: 20K-1610