



20F-0292_BCS 4D_COAL LAB FINAL

Saad Abdur Razzaq



TASK 1:

(THIS TASK HAS BOTH PARTS PART A AND PART B MERGED TO ONE SINGLE PROGRAM)

Code:

```
INCLUDE Irvine32.inc

.data
arr dword 1, 2, 3, 4, 5
sum dword ?
msg1 byte "                ****SUM OF ALL ELEMENTS OF ARRAY AND REVERSAL
OF ARRAY****",0
dash byte "                <-----"
----->",0
str1 byte "---> PART A:",0
msg0 byte "5 4 3 2 1 ",0
str2 byte "---> PART B:",0
msg2 byte "Elements of Array are: ",0
msg3 byte "1 2 3 4 5 ",0
msg4 byte "The Sum of all Array elements are: ",0
msg5 byte "Original Array: ",0
msg6 byte "Reversed Array: ",0
arrSize DWORD SIZEOF ARR
.code
main PROC
call crlf
mov edx, offset msg1
call writestring
    call crlf
    call crlf
mov edx, offset STR1
call writestring

    call crlf
    call crlf
mov edx, offset msg2
call writestring

mov edx, offset msg3
call writestring

    call crlf
    call crlf
mov edx, offset msg4
call writestring
mov esi,offset arr
mov ecx,5
mov eax,0
11:
```

```

add eax, ecx
loop l1
call writedec

        call crlf
        call crlf
        call crlf
mov edx, offset dash
call writestring
        call crlf
        call crlf
        call crlf

        mov edx, offset STR2
call writestring

        call crlf
        call crlf
mov edx, offset msg5
call writestring
mov edx, offset msg3
call writestring
        call crlf
        call crlf
        mov edx, offset msg6
call writestring
mov esi,offset arr
mov edx, offset msg0
call writestring
call crlf
        call crlf
        call crlf

        call WaitMsg
exit
main ENDP
END main

```

Outputs:

```
D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

****SUM OF ALL ELEMENTS OF ARRAY AND REVERSAL OF ARRAY****

---> PART A:
Elements of Array are: 1 2 3 4 5
The Sum of all Array elements are: 15

<----->

---> PART B:
Original Array: 1 2 3 4 5
Reversed Array: 5 4 3 2 1

Press any key to continue...
```

TASK 2:

Code:

```
INCLUDE Irvine32.inc

.data
arr dword 3, 4, 8, 20, 25, 30
str1 byte "
msg0 byte "3 4 8 20 25 30",0
msg1 byte "arr[",0
msg2 byte "]",0
msg3 byte " =>",0
msg4 byte "Original pathed array: ",0
msg5 byte "Special pathed array: ",0
.code
main PROC
call crlf
mov edx, offset str1
call writestring
call crlf
call crlf
mov edx, offset msg4
call writestring
mov edx, offset msg0
call writestring

call crlf
call crlf
mov edx, offset msg5
call writestring
call crlf
call crlf
mov ecx, 6
L1:
loop L1

****DISPLAY ARRAY OF A SPECIFIC FORMAT****,0
```

```
mov edx, offset msg1
call writestring
mov eax, 0
call writedec
mov edx, offset msg2
call writestring
mov edx, offset msg3
call writestring
mov eax, 3
call writedec
call crlf
```

```
mov edx, offset msg1
call writestring
mov eax, 1
call writedec
mov edx, offset msg2
call writestring
mov edx, offset msg3
call writestring
mov eax, 4
call writedec
call crlf
```

```
mov edx, offset msg1
call writestring
mov eax, 2
call writedec
mov edx, offset msg2
call writestring
mov edx, offset msg3
call writestring
mov eax, 8
call writedec
call crlf
```

```
mov edx, offset msg1
call writestring
mov eax, 3
call writedec
mov edx, offset msg2
call writestring
mov edx, offset msg3
call writestring
mov eax, 20
call writedec
call crlf
```

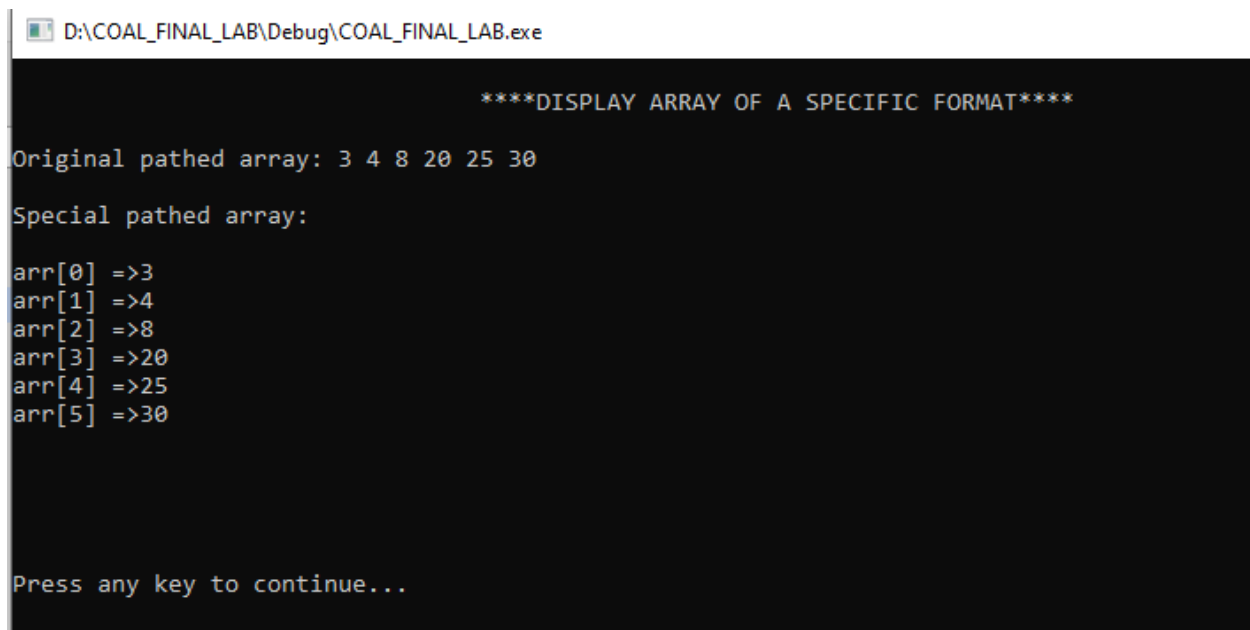
```
mov edx, offset msg1
call writestring
mov eax, 4
call writedec
mov edx, offset msg2
call writestring
mov edx, offset msg3
call writestring
mov eax, 25
call writedec
```

```
call crlf

mov edx, offset msg1
call writestring
mov eax, 5
call writedec
mov edx, offset msg2
call writestring
mov edx, offset msg3
call writestring
mov eax, 30
call writedec
call crlf
    call crlf
    call crlf

    call WaitMsg
exit
main ENDP
END main
```

Outputs:



```
D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

****DISPLAY ARRAY OF A SPECIFIC FORMAT****

Original pathed array: 3 4 8 20 25 30

Special pathed array:

arr[0] =>3
arr[1] =>4
arr[2] =>8
arr[3] =>20
arr[4] =>25
arr[5] =>30

Press any key to continue...
```

TASK 5:

Code:

```

INCLUDE Irvine32.inc
.data

str1 byte "                ****SUBTRACTION OF TWO BINARY INTEGERS****",0
num1 dword ?
num2 dword ?
msg1 byte "Enter First binary Number: ",0
msg2 byte "Enter Second binary Number: ",0
final byte "The Subtraction of two Binary integers is: ",0
.code
main PROC

    call crlf

    mov edx, offset str1
    call writestring

    call crlf

    mov edx, offset msg1
    call writestring

    call readdec

    mov num1, eax

    call crlf
    call crlf

    mov edx, offset msg2
    call writestring

    call readdec
    mov num2, eax
    call crlf
    mov edx, offset final
    call writestring
    CALL Extended_Sub
    call writebin

        call crlf
        call crlf
        call crlf

        call WaitMsg

    exit
main ENDP


Extended_Sub PROC
    mov eax, num1
    mov ebx, num2
    sub eax, ebx
    RET
Extended_Sub endp
END main

```

Outputs:

D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
****SUBTRACTION OF TWO BINARY INTEGERS****
Enter First binary Number: 10101010110101

Enter Second binary Number: 1101010011

The Subtraction of two Binary integers is: 1011 1110 0101 1111 1110 1011 1010 0101

Press any key to continue..._
```

D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
****SUBTRACTION OF TWO BINARY INTEGERS****
Enter First binary Number: 1111000101110

Enter Second binary Number: 101100110

The Subtraction of two Binary integers is: 1111 1001 1111 1001 0101 0101 1011 0010

Press any key to continue..._
```

TASK 6:

Code:

```
INCLUDE Irvine32.inc

.data
msg1 byte "****EAX EXPRESSION EVALUATION****",0
STR1 BYTE "Expression for evaluation is: ",0
STR2 BYTE "EAX = (30 / 6) + (10 - 5) + (3 x 5)",0
str3 byte "Answer is: ",0
num1 dword ?
num2 dword ?
num3 dword ?
num4 dword ?
.code
main PROC
call crlf
mov edx, offset msg1
call writestring
call crlf
```



```

        call crlf
mov edx, offset STR1
call writestring
        call crlf
        call crlf
mov edx, offset str2
call writestring
        call crlf
        call crlf

mov edx, offset str3
call writestring


    mov eax,30
    mov ebx,6
    mov edx, 0
div ebx
mov num1,eax ;(30/6)
mov eax, 0
    mov ebx,0
    mov eax, 10
    mov ebx, 5
    sub eax, ebx
    mov num2, eax ;(10-5)
mov eax, 0
    mov ebx,0
    mov eax, 3
    mov ebx, 5
    mul ebx
    mov num3, eax ;(3x5)
mov eax, 0
    mov ebx,0
    mov eax,num1
    mov ebx, num2
    add eax, ebx
    mov num4, eax ;(30/6)+(10-5)
mov eax, 0
    mov ebx,0
    mov eax, num4
    mov ebx, num3
    add eax,ebx ;(30/6)+(10-5)+(3x5)
call writedec
call crlf
        call crlf
        call crlf

        call WaitMsg

exit
main ENDP
END main

```

Outputs:

D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
****EAX EXPRESSION EVALUATION****  
  
Expression for evaluation is:  
  
EAX = (30 / 6) + (10 - 5) + (3 x 5)  
  
Answer is: 25  
  
Press any key to continue...
```

TASK 8:


Code:

```
INCLUDE Irvine32.inc  
  
.data  
str1 byte "                ****SUM OF TWO NUMBERS****",0  
num1 dword ?  
num2 dword ?  
msg1 byte "Enter First Number (Range 0-4): ",0  
msg2 byte "Enter Second Number (Range 0-4): ",0  
sum byte "The Sum of two numbers is: ",0  
.code  
main PROC  
    call crlf  
    mov edx, offset str1  
    call writestring  
    call crlf  
    mov edx, offset msg1  
    call writestring  
    call readdec  
    mov num1, eax  
    mov eax,0  
  
    call crlf  
    call crlf  
    mov edx, offset msg2  
    call writestring  
    call readdec  
    mov num2, eax  
    mov eax,0  
  
    call crlf
```

```
call crlf
mov edx, offset sum
call writestring
mov eax, num1
mov ebx, num2
add eax, ebx
call writedec
call crlf
        call crlf
        call crlf

        call WaitMsg
exit
main ENDP
END main
```

Outputs:

 D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
*****SUM OF TWO NUMBERS*****
Enter First Number (Range 0-4): 1

Enter Second Number (Range 0-4): 6

The Sum of two numbers is: 7

Press any key to continue...
```

D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe


```
*****SUM OF TWO NUMBERS*****  
Enter First Number (Range 0-4): 55  
  
Enter Second Number (Range 0-4): 100  
  
The Sum of two numbers is: 155  
  
Press any key to continue...
```

TASK 9:


Code:

```
INCLUDE Irvine32.inc  
  
.data  
str1 byte "*****DECIMAL TO BINARY CONVERSION*****",0  
num1 word ?  
msg1 byte "Enter Number (Range 0-15): ",0  
msg2 byte "The Binary of ",0  
msg3 byte " is: ",0  
.code  
main PROC  
  
    call crlf  
  
    mov edx, offset str1  
    call writestring  
  
    call crlf  
  
    mov edx, offset msg1  
    call writestring  
  
    call readdec  
  
    mov num1, ax  
  
    call crlf  
    call crlf  
  
    mov edx, offset msg2
```

```
call writestring  
  
call writedec  
  
mov edx, offset msg3  
call writestring  
  
call writebin  
  
    call crlf  
        call crlf  
        call crlf  
  
    call WaitMsg  
exit  
main ENDP  
END main
```

 D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
*****DECIMAL TO BINARY CONVERSION*****  
Enter Number (Range 0-15): 12  
  
The Binary of 12 is: 0000 0000 0000 0000 0000 0000 0000 1100  
  
Press any key to continue..._
```

 D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
*****DECIMAL TO BINARY CONVERSION*****  
Enter Number (Range 0-15): 9  
  
The Binary of 9 is: 0000 0000 0000 0000 0000 0000 0000 1001  
  
Press any key to continue...
```

D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
****DECIMAL TO BINARY CONVERSION****  
Enter Number (Range 0-15): 15  
  
The Binary of 15 is: 0000 0000 0000 0000 0000 0000 0000 1111  
  
Press any key to continue...
```

TASK 10:

Code:

```
INCLUDE Irvine32.inc  
  
.data  
str1 byte "****GRADING SYSTEM****",0  
num1 dword ?  
num2 dword ?  
msg1 byte "Enter Your test score (Range 0-100): ",0  
agrade byte "You achieved A grade ",0  
bgrade byte "You achieved B grade ",0  
cgrade byte "You achieved C grade ",0  
dgrade byte "You achieved D grade ",0  
egrade byte "You achieved E grade ",0  
fgrade byte "You achieved F grade ",0  
.code  
main PROC  
call crlf  
mov edx, offset str1  
call writestring  
call crlf  
mov edx, offset msg1  
call writestring  
call readdec  
mov num1, eax  
mov eax,90  
MOV EBX,100  
CMP num1, EBX  
jbe a2  
CMP num1, EaX
```

```
jae a1
a1:
mov edx, offset agrade
call writestring
call crlf
a2:
```

```
mov num1, eax
mov eax,80
MOV EBX,89
CMP num1, EBX
jbe b2
CMP num1, EaX
jae b1
b1:
mov edx, offset bgrade
call writestring
call crlf
b2:
```


```
mov num1, eax
mov eax,70
MOV EBX,79
CMP num1, EBX
jbe c2
CMP num1, EaX
jae c1
c1:
mov edx, offset cgrade
call writestring
call crlf
c2:
```

```
mov num1, eax
mov eax,60
MOV EBX,69
CMP num1, EBX
jbe d2
CMP num1, EaX
jae d1
d1:
mov edx, offset dgrade
call writestring
call crlf
d2:
```


```
mov num1, eax
mov eax,0
MOV EBX,59
CMP num1, EBX
jbe f2
CMP num1, EaX
jae f1
f1:
mov edx, offset fgrade
call writestring
call crlf
f2:
```

```
        call crlf
        call crlf


        call WaitMsg
    exit
main ENDP
END main
```

 D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
                ****GRADING SYSTEM****
Enter Your test score (Range 0-100): 85
You achieved B grade
```

 D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
                ****GRADING SYSTEM****
Enter Your test score (Range 0-100): 75
You achieved C grade
```

 D:\COAL_FINAL_LAB\Debug\COAL_FINAL_LAB.exe

```
                ****GRADING SYSTEM****
Enter Your test score (Range 0-100): 2
You achieved F grade
```