COAL LAB 9

23K-0057 BAI-4A

Q1.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc

.data
var1 DWORD 5
var2 DWORD 6
var3 DWORD 7
.code
main PROC
push var3
push var2
push var1
call ThreeProd
call writedec
exit
main ENDP
ThreeProd PROC
push ebp
mov ebp, esp
mov eax, [ebp + 12]
mul DWORD PTR [ebp + 8]
mul DWORD PTR [ebp + 16]
pop ebp
ret
ThreeProd ENDP
END main
```

```
210
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q2.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
array DWORD 5,10,15,20,25,30,35,40,45,50,60,70,80,90,100,110,120,130,140,150
count DWORD LENGTHOF array
min BYTE "Minimum value in array: ", 0
max BYTE "Maximum value in array: ", 0

.code
main PROC
push OFFSET array
push count
call MinMaxArray
exit
main ENDP
MinMaxArray PROC
push ebp
mov ebp, esp
mov esi, [ebp + 12]
mov ecx, [ebp + 8]
cmp ecx, 0
je L2
mov eax, [esi]
L1:
mov ebx, [esi]
cmp ebx, eax
jg skip
```

```
mov eax, ebx
skip:
add esi, 4
loop L1
L2:
mov edx, OFFSET min
call WriteString
call WriteDec
call Crlf
mov esi, [ebp + 12]
mov ecx, [ebp + 8]
cmp ecx, 0
je done
mov eax, [esi]
L4:
mov ebx, [esi]
cmp ebx, eax
jl skip2
mov eax, ebx
skip2:
add esi, 4
loop L4
done:
mov edx, OFFSET max
call WriteString
call WriteDec
pop ebp
ret 8
MinMaxArray ENDP
END main
```

```
Minimum value in array: 5
Maximum value in array: 150
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q3.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
input BYTE "Enter number: ", 0
sq BYTE "Square of number is: ", 0

.code
main PROC
call LocalSquare
exit
main ENDP
LocalSquare PROC
mov edx, OFFSET input
call WriteString
call ReadDec
push ebp
mov ebp, esp
mov DWORD PTR [ebp - 4], eax
mul eax
mov edx, OFFSET sq
call WriteString
call WriteDec
leave
ret
LocalSquare ENDP
END main
```

```
Enter number: 4
Square of number is: 16
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q4.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
msg1 BYTE "Enter number: ",0
msg2 BYTE " is prime",0
msg3 BYTE " is not a prime number",0
msg4 BYTE "All the numbers are prime",0
msg5 BYTE " is the largest prime number",0
n1 DWORD ?
n2 DWORD ?
n3 DWORD ?
n4 DWORD ?
num DWORD ?
count DWORD 0

.code
main PROC
mov edx,offset msg1
call writestring
call readdec
mov n1,eax
mov edx,offset msg1
call writestring
call readdec
mov n2,eax
mov edx,offset msg1
call writestring
call readdec
mov n3,eax
mov edx,offset msg1
call writestring
call readdec
mov n4,eax

call CheckPrime
exit
main ENDP
CheckPrime PROC
push n4
push n3
push n2
push n1
mov ebx,4
l1:
pop eax
mov num,eax
cmp eax, 2
jl notprimee
cmp eax, 2
je primee
mov ecx, 2
mov esi,eax
check_loop:
mov eax,esi
mov edx, 0
div ecx
cmp edx, 0
je notprimee
inc ecx
```

```
cmp ecx, esi
jl check_loop
primee:
mov eax,num
call writedec
mov edx, offset msg2
call writestring
call crlf
inc count
dec ebx
cmp ebx, 0
je countt
jmp l1
notprimee:
mov eax,num
call writedec
mov edx, offset msg3
call writestring
call crlf
dec ebx
cmp ebx, 0
je countt
jmp l1
countt:
mov eax, count
cmp eax, 4
jne returnn
mov edx, offset msg4
call writestring
call LargestPrime
returnn:
ret
CheckPrime ENDP

LargestPrime PROC
call crlf
push n4
push n3
push n2
push n1
mov eax , -999
mov ecx,4
l2:
pop ebx
cmp eax,ebx
jg exitl2
mov eax,ebx
exitl2:
loop l2
call writedec
mov edx,offset msg5
call writestring
ret
LargestPrime ENDP
END main
```

```
Enter number: 2
Enter number: 3
Enter number: 5
Enter number: 11
2 is prime
3 is prime
5 is prime
11 is prime
All the numbers are prime
11 is the largest prime number
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q5.

```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
arr DWORD 80,90,20,50,40
sizee DWORD LENGTHOF arr
original BYTE "Original Array: ", 0
sorted BYTE "Sorted Array: ", 0

.code
main PROC
mov edx, OFFSET original
call WriteString
mov ecx, sizee
lea esi, arr
call Displayarr
call Crlf
lea esi, arr
push esi
push sizee
call BubbleSort
mov edx, OFFSET sorted
call WriteString
mov ecx, sizee
lea esi, arr
call Displayarr
call Crlf
exit
main ENDP
BubbleSort PROC
push ebp
mov ebp, esp
mov ecx, [ebp+8]
mov esi, [ebp+12]
dec ecx
outer:
mov eax, 0
mov edi, esi
mov ebx, ecx
inner:
mov eax, [edi]
mov edx, [edi+4]
cmp eax, edx
jle skip
mov [edi], edx
mov [edi+4], eax
mov eax, 1
skip:
add edi, 4
dec ebx
```

```
jnz inner
test eax, eax
jz done
dec ecx
jnz outer
done:
pop ebp
ret
BubbleSort ENDP
Displayarr PROC
mov eax, 0
mov ebx, ecx
display:
mov eax, [esi]
call WriteDec
mov al, ' '
call WriteChar
add esi, 4
dec ebx
jnz display
ret
Displayarr ENDP
END main
```

```
Original Array: 80 90 20 50 40
Sorted Array: 20 40 50 80 90

C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```