COAL LAB 11

23K-0057 (BAI-4A)

Q1.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
.code
main PROC
mov ecx, 0
mov eax, 4
mov ebx, eax
shl eax, 4
add ecx, eax
mov eax, ebx
shl eax, 2
add ecx, eax
mov eax, ebx
add ecx, eax
mov eax, ecx
call WriteDec
exit
main ENDP
END main
```

```
84
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q2.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
.code
main PROC
mov eax, 0
mov ax, -128
call WriteBin
shl eax, 16
call crlf
call WriteBin
sar eax, 16
call crlf
call WriteBin
call crlf
call WriteInt
exit
main ENDP
END main
```

```
0000 0000 0000 0000 1111 1111 1000 0000
1111 1111 1000 0000 0000 0000 0000 0000
1111 1111 1111 1111 1111 1111 1000 0000
-128
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q3.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
bMinutes byte ?
.code
main PROC
```

```
mov ax, 1111111111111111b
mov bx,ax
shr bx,5
and bl,00111111b
mov bMinutes,bl
movzx eax,bl
call WriteBin
exit
main ENDP
END main
```

```
0000 0000 0000 0000 0000 0000 0011 1111
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q4.

```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
.code
main PROC
mov eax, 10010011b
mov ebx, 01010011b
shr ax,1
rcr bx,1
call WriteBin
call crlf
mov ax,bx
call WriteBin

call crlf
call crlf

;with shrd
mov eax, 10010011b
mov ebx, 01010011b
shrd ax,bx,1
call WriteBin
call crlf
exit
main ENDP
END main
```

```
0000 0000 0000 0000 0000 0000 0100 1001
0000 0000 0000 0000 1000 0000 0010 1001

0000 0000 0000 0000 1000 0000 0100 1001

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

Q5.

```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
val1 dword 10
val2 dword 5
val3 dword 5
temp1 dword ?
temp2 dword ?
msg byte "Val1 = ",0
.code
main PROC
mov eax,val2
cdq
idiv val3
```

```
mov temp1,eax
mov eax,val1
cdq
idiv val2
mov temp2,eax
mov eax,temp1
imul eax,temp2
mov val1,eax
mov edx,offset msg
call WriteString
call WriteDec
exit
main ENDP
END main
```

```
Val1 = 2
C:\Users\DELL\source\repos\Project1\Debug\Project1.exe
```

Q6.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
num1 dword 12345678h ; lower
num2 dword 00000001h ; upper
num11 dword 87654321h ; lower
num22 dword 00000002h ; upper
result1 dword ?
result11 dword ?
.code
Extended_Add PROC
mov eax, num1
add eax, num11
mov result1, eax
mov eax, num2
adc eax, num22
mov result11, eax
ret
Extended_Add ENDP

main PROC
call Extended_Add
mov eax, result1 ; lower
call WriteHex
call crlf
mov eax, result11 ; upper
call WriteHex
call crlf
exit
main ENDP
END main
```

```
99999999
00000003

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

Q7.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
prompt byte "Enter number : ",0
primeMsg byte " Prime ",0
NprimeMsg byte " Not Prime ",0
num dword 0
```

```
.code
IsPrime PROC
mov edx, offset prompt
call WriteString
call ReadDec
cmp eax, 1
je returning
cmp eax, 2
je Primee
mov ecx, eax
mov num,2
sub ecx, 2

i:
push eax
mov edx ,0
div num
cmp edx,0
pop eax
jz Not_prime
add num,1
loop i

Primee:
call WriteDec
mov edx, offset primeMsg
call WriteString
call crlf
jmp looping
Not_prime:
call WriteDec
mov edx, offset NprimeMsg
call WriteString
call crlf
jmp looping
looping:
call IsPrime
returning:
ret
IsPrime ENDP
main PROC
call IsPrime
exit
main ENDP
END main
```

```
Enter number : 10
10 Not Prime
Enter number : 4
4 Not Prime
Enter number : 2
2 Prime
Enter number : 1

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

Q8.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
plaintext byte "abcd efgh ijkl mnop", 0
key byte 2, 4, 1, 0, 3, 5, 2, 4, 4, 6
str1 byte "Encrypted Msg: ",0
.code
```

```
main PROC
mov ecx, LENGTHOF plaintext
mov esi, 0
mov edi, 0

l1:
push ecx
cmp edi, LENGTHOF key
jl continuee
mov edi, 0
continuee:
mov cl, key[edi*TYPE key]
cmp eax, 0
jg RightRotate
jl LeftRotate
jmp _comp
LeftRotate:
rol plaintext[esi*TYPE plaintext], cl
jmp _comp
RightRotate:
ror plaintext[esi*TYPE plaintext], cl
_comp:
pop ecx
inc esi
inc edi
loop l1

call crlf
mov edx, offset str1
call WriteString
mov edx, offset plaintext
call WriteString
call crlf
exit
main ENDP
END main
```

```
Encrypted Msg: X&▒d+ÖvåÇZª╡lk¢÷

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

Q9.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
msg BYTE "Result: ",0
.code
main PROC
mov eax, 14 ; multiplier
mov ebx, 3 ; multiplicand
mov edx, 0
call BitwiseMultiply
exit
main ENDP

BitwiseMultiply PROC
L1:
cmp eax, 0
je donee
test eax, 1
jz shiftt
add edx, ebx
shiftt:
shr eax, 1
shl ebx, 1
```

```
jmp L1
donee:
mov eax, edx
mov edx, offset msg
call crlf
call WriteString
call WriteDec
call crlf
ret
BitwiseMultiply ENDP
END main
```

```
Result: 42

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

Q10.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc
.data
str1 byte "Enter the first number: ",0
str2 byte "Enter the second number: ",0
str3 byte "GCD of both the numbers is ",0
.code
GCD PROC
L1:
cmp ebx, 0
jbe endd
cdq
div ebx
mov eax, ebx
mov ebx, edx
JMP L1
endd:
ret
GCD ENDP
main PROC
call crlf
mov edx, offset str1
call WriteString
call ReadDec
mov ebx, eax
mov edx, offset str2
call WriteString
call ReadDec
mov edx, eax
mov eax, ebx
mov ebx, edx
call GCD
mov edx, offset str3
call WriteString
call WriteDec
call crlf
exit
main ENDP
END main
```

```
Enter the first number: 12
Enter the second number: 6
GCD of both the numbers is 6
```

Q11.
```
TITLE My First Program (Test.asm)
INCLUDE Irvine32.inc

DECIMAL_OFFSET equ 5
.data
decimal_one BYTE "100123456789765", 0
.code
main PROC
call WriteScaled
exit
main ENDP

WriteScaled PROC
mov esi, OFFSET decimal_one
mov ecx, 0
mov edi, esi
mov edx, 0
getLength:
mov al, [edi]
inc edi
inc edx
cmp al, 0
jne getLength
sub edx, DECIMAL_OFFSET
printt:
mov al, [esi]
call WriteChar
inc esi
inc ecx
cmp ecx, edx
jne skippingDec
mov al, '.'
call WriteChar
skippingDec:
cmp BYTE PTR [esi], 0
jne printt
ret
WriteScaled ENDP
exit
END main
```