Lab: 5

Fahid Imran

23i-0061

COAL

Instructor

Mr. Sulaman Saboor

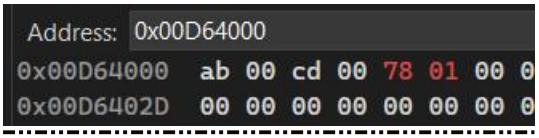
Fast NUCES Islamabad Campus

Task1:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data
num1 WORD 0abh
num2 WORD 0cdh
result WORD ?
.code

main PROC

mov ax, num1
mov bx, num2

add ax,bx
mov result , ax

main endp
end main
```

Output:



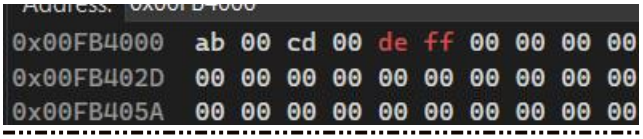
Task2:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data
```

```
num1 WORD 0abh
num2 WORD 0cdh
result WORD ?
.code

main PROC

mov ax, num1
mov bx, num2

sub ax,bx
mov result , ax

main endp
end main
```

Output:



Task3:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data
num1 DWORD 036h
num2 DWORD 006h
result DWORD ?
.code

main PROC

mov eax, num1
mov ebx, num2
mov edx , 0
mul ebx
mov result , eax
```

```
main endp
end main
```

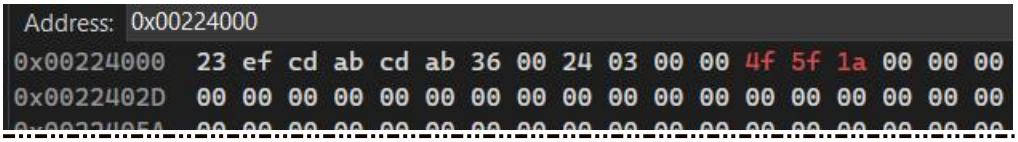
Output:

Task4:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data
num1 DWORD 0abcdef23h
num2 DWORD 036abcdh
result DWORD ?
remainder DWORD ?
.code

main PROC

mov eax, num1
mov ebx, num2
mov edx , 0
div ebx
mov result , eax
mov remainder, edx
main endp
end main
```

Output:



Task5:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data

String Byte 'Fahid Imran'
var1 Word ?
var2 Word ?
var3 Word ?


.code

main PROC
mov ax , sizeof String
mov var1, ax
mov ax , lengthof String
mov var2, ax
mov ax , Type String
mov var3, ax

main endp
end main
```

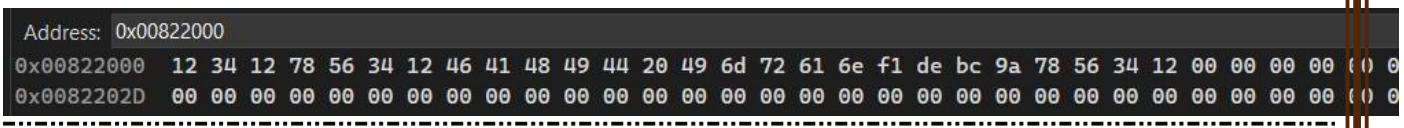
Output:

Task6:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data
num1 Byte 012h
num2 word 01234h
num3 DWORD 012345678h
atr byte 'FAHID Imran'
num4 QWORD 0123456789abcdef1h

.code

main PROC

main endp
end main
```

Output:



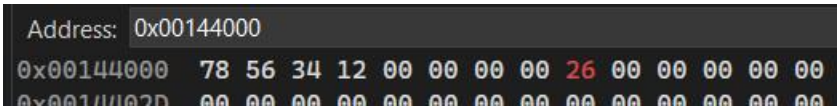
Task7:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data
```

```
var1 dword 012345678h
var2 dword ?
var3 Byte ?

.code

main PROC
mov eax , [var1+2]
mov ebx , var1
mov edx, 0
div bl
mov var3, al

main endp
end main
```

Output:



Task8:

Code:

```
.386
.model flat, stdcall
.stack 4098

.data

var1 dword 012345678h
var2 dword ?
var3 Word ?

.code

main PROC
mov eax , [var1+3]
mov ebx , var1
mov edx, 0
mul bl
```

```
mov var3, ax

main endp
end main
```

Output: