

# **:: ASSIGNMENT # 3 ::**

**ID: 11533**

**Name: Ashir Ali Khan**

**Subject: Microprocessor and  
Assembly Language**

**Teacher: Sir Muhammad Amin**



**Iqra National University**

11593

Date: ...../...../20.....

①

M  T  W  T  F  S

2020/5/20 10:21

# Assignment #3

Q<sub>1</sub> Answer:

-35d, DDh, 3350, 1101101b

Q<sub>2</sub> Answer:
~~→ 62E04~~  
 $(5+1)(-2+3) * 2 \text{ mod } 5 = 2$ 
Q<sub>3</sub> Answer:

-6.2E + 04

Q<sub>4</sub> Answer:

Include:

inserts ~~and~~ source code from the sourcefile given by filename into current sourcefile during assembly.

Syntax

INCLUDE filename

.386

Enables assembly of nonprivileged instructions for the 80386 processor; disables assembly of instruction introduced with later process. Also enables 80387 instruction.

Syntax

.386

REDMI NOTE 8  
AI QUAD CAMERA

11533

Date: ...../...../20.....

(2)

MOTOWOTOP 3

2020/5/20 10:21

### • MODEL:

This tells the assembler which memory model to use in 32-bit program. We use the flat memory model which is associated with the processor's protected model.

### • STACK:

The stack directive tells how many bytes of memory to reserve for the runtime stack. 4096 happens to correspond to the size of a memory page in this processor system for managing memory.

### • PROTO:

The proto directives by using the invoke directives.

### • DATA:

The DATA directives creates near a data segment. This segment contains the ~~freq~~ frequency and data for your program. Data segment can occupy upto 64K in MS-Dos or upto 512 mega-byte under flat model in window NT.

### • CODE:

code  
main PROC  
it is the beginning of the code area of the program.

11533

Date: ...../...../20.....

③

M  T  W  T  F  S**PROC:**

~~Mark~~ MASK starts and ends of a procedure block called label. The statement in the block can be called with the call instruction or INVOKE directives.

**ENDP:**

Mark the end of procedure name previously begun with ~~proc~~ PROC:

**END:**

Directives for END of file command. That's end of file here as you are using "main" you have to end it with:

**Q5 Answer:**

The exit statement (indirectly) ~~halts~~ call a predefined MS window function that halts the program. The ENDP directive marks the end of the ~~file~~ main procedure. The END main direct. marks the last line of the program to be assembled.

**Q6 Answer:**

An integer preferably 0.

**Q7 Answer:**

A calling convention determines how parameters are passed to sub-routine and how the stack is restored after the subroutine call.

2020/5/20 10:21

REDMI NOTE 8  
AI QUAD CAMERADass  
Note

11533

Date: ...../...../20.....

④

M T W T F S S

2020/5/20 10:21

Q8

Answer:

The main output produced by assembler on input assembly language source file is the translation of the file into an object file in (ELF). ELF files produced by the assembler are relocatable files that hold code and/or data. They are input files for the linker.

Q9

Answer:

A monolithic kernel runs all the operating system ~~introduction~~ instruction in the same address space for speed. A monokernel run most processes in user space for modularity. This central component of a computer system is responsible for running or executing programs.

Q10

Answer:

signed 8 bit : var3 SBYTE #+127  
 unsigned 8 bit : var2 BYTE # 0  
 signed 16 bit : var1 SWORD #-32768  
 unsigned 16 bit : var4 WORD #65535  
 signed 32 bit : var5 SDWORD #-214768  
 unsigned 32 bit : var6 DWORD #12345678

(D.T.O)

REDMI NOTE 8  
AI QUAD CAMERA

Notes

11533

Date: ...../...../20.....

5

M  T  W  T  F  S

2020/5/20 10:21

Q11 Answer:-

val DWORD 12345678h

0000:	78
0001:	56
0002:	34
0003:	12

Q12 Answer:

val DWORD 87654321h

21h

43h

65h

87h

Q13 Answer:

Yes, you can. The assembler does not check the number's sign.

Q14 Answer:

Array DWORD 120(?)

Q15 Answer:

myArray BYTE 'A', 'B', 'C', 'D', 'E'

Q16 Answer:

Array WORD 10, 20, 30.

REDMI NOTE 8  
AI QUAD CAMERA

Dam  
Note

11533

Date: ...../...../20.....

⑥

MOTOWOTOFOS

2020/5/20 10:21

Q<sup>17</sup> Answer:

myColor BYTE "blue", 0

Q<sup>18</sup> Answer:

An integer literal, such as 35, has no direct meaning to someone reading the program's source code. Instead, a symbolic constant such as MIN\_CAPACITY can be assigned an integer value, and is self documenting.

Q<sup>19</sup> Answer:

ArraySize = (\$-myArray)

Q<sup>20</sup> Answer:ArraySize = (\$-myArray)/4 ~~OR~~

OR

ArraySize = (\$-myArray)/TYPE DWORD

Q<sup>21</sup> Answer:

Program:

.386

-model flat,stdcall

-stack 4096

ExitProcess PROTO, dwExitCode:DWORD

SUBN=0

MON=1

TUE=2

WED=3

REDMI NOTE 8  
AI QUAD CAMERAParsi  
Notes

11533

Date: ...../...../20.....

MOTOWTFFS

2020/5/20 10:21

```
THU=4
FRI=5
SAT=6
.data
    daysArray BYTE SUN,MON,TUE,WED,THU,FRI,SAT
.code
    main PROC main PROC
    INVOKE ExitProcess,0
    main ENDP
END main
```

Q2 Answer:

Program:

.386

.model flat,stdcall

.stack 4096

ExitProcess PROTO, dwExitCode:DWORD

str1 EQUK "Ashir",0

str2 EQUK "Ali",0

str3 EQUK "Khan",0

.data

first BYTE str1

second BYTE str2

third BYTE str3

.code

main PROC

~~Invoke~~ INVOKE ExitProcess,0

main ENDP

END main

REDMI NOTE 8  
AI QUAD CAMERA

Darsi  
Notes



11533

Date: ...../...../20.....

②

MOTOWOTOP

2020/5/20 19:22

Q 23 Answer:

### Equal Sign directive:

The equal sign directive associates a symbol name with an integer expression

This syntax is

name = Expression .

- Expression is a 32-bit integer
- May be defined
- Name is called a symbolic constant.

### EQU - directive:

- Define a symbol as either an integer or text expression.
- Cannot be redefined

Q 24 Answer:

### Zero Operands:

stc ; set carry flag

### One Operand:

inc eax ; register

inc myByte ; memory

### Two Operands:

add ebx, eax ; register, register

sub myByte, 25 ; memory, constant

add eax, 36x25 ; register, constant

expression.

REDMI NOTE 8  
AI QUAD CAMERA

asi  
Notes

Q. 25 Answer:

A sourcefile is given as input to the assembler.

A listing file has additional text that will not assemble. It is a file that is created by the assembler.

Q. 26 Answer:

Code example:

The two instructions have different opcodes

add eax, 5

add edx, 5

Q. 27 Answer:

Data labels exist in the data segment as variable offsets

Code labels are in the code segment and are offsets for transfer of control instructions.

Q. 28 Answer:

- ① [Label:]
- ② mnemonic
- ③ [operand]
- ④ [Comment]

(P.T. 0)

11533

Date: ...../...../20.....

10

MOTOWTOPES

2020/5/20 10:22

Q<sup>29</sup> Answer:

Comment!  
First line comment  
second line comment

Q<sup>30</sup> Answer:

You do not use numeric addresses for variables because the addresses would change if new variable were inserted before the existing ones.

Q<sup>31</sup> Answer:

The .EXE format can have multiple code, data and stack segments.

Q<sup>32</sup> Answer:

Program:

```
.386
.model flat, stdcall
.stack 4096
ExitProcess PROTO, dwExitCode: DWord
.code
main PROC
    mov eax, 3h
    mov ebx, 8h
    mov ecx, 1h
    mov edx, 8h
```

REDMI NOTE 8  
AI QUAD CAMERA

Parsha  
Notes

11533

Date: ...../...../20.....



MOTO WOTOFOS

```
add eax, ebx
add ecx, edx
sub eax, ecx
INVOKE ExitProcess, 0
main ENDP
END main.
```

2020/5/20 10:22

REDMI NOTE 8  
AI QUAD CAMERA

