|  |  |
| --- | --- |
|  | **Department of Computer Science**  **Faculty of Computing**  **UNIVERSITI TEKNOLOGI MALAYSIA** |

|  |  |
| --- | --- |
| **SUBJECT NAME:** | **COMPUTER ORGANIZATION AND ARCHITECTURE** |
| **SUBJECT CODE:** | **SECR 2033** |
| **SEMESTER:** | **1 – 2020/2021** |
| **LAB TITLE:** | **Programming 2: Arithmetic Equations & Operations** |
| **STUDENT INFO :** | **Name: HAM JING YI**  **Metric No: A19EC0048** |
| **SUBMITTED DATE:** | 23-4-2020 |

|  |  |
| --- | --- |
| **COMMENTS:** |  |
|  |  |

**Part B: Arithmetic Equation Coding in Assembly Language (Equation to Code)**

**MARKS:**

**\_\_\_\_\_\_\_\_**

1. Execute the program below. Determine output of the program by inspecting the content of the related registers. Fill in Table 1 with the content of each register or variable on every LINE, in **Hexadecimal** (as per the output). Please complete the comments for every LINE.

INCLUDE Irvine32.inc

.data

var1 word 1

var2 word 9

.code

main PROC

mov ax, var1 ; LINE1

mov bx, var2 ; LINE2

xchg ax, bx ; LINE3

mov var1, ax ; LINE4

mov var2, bx ; LINE5

call DumpRegs

exit

main ENDP

END main

**Answer:**

|  |  |  |
| --- | --- | --- |
|  |  | **Table 1** |
| LINE1 | **AX = 0001h** | Move the value of var1 (1d) into register |
| **var1 = 0001h** | AX |
|  |
| LINE2 | **BX =0009h** | Move the value of var2 (9d) into register |
| **var2 =0009h** | BX |
|  |  |
| LINE3 | **AX =0009h** | Exchange the value of register AX and register BX |
| **BX =0001h** |  |
|  |  |
| LINE4 | **AX =0009h** | Move the value of register AX (9d) into var1 |
| **var1 =0009h** |  |
|  |  |
| LINE5 | **BX =0001h** | Move the value of register BX (1d) into var2 |
| **var2 =0001h** |  |
|  |  |

2. Execute the program below. Determine output of the program by inspecting the content of the related registers and watches. Fill in Table 2 with the content of each register or variable on every LINE, in **Hexadecimal** (as per the output). Please complete the comments for every LINE.

**Arithmetic expression: Rval = (-Xval + (Yval – Zval)) + 1**

include irvine32.inc

.data

Rval DWORD ?

Xval DWORD 26

Yval DWORD 30

Zval DWORD 40

.code

main proc

mov eax,Xval ; LINE1

neg eax ; LINE2

mov ebx,Yval ; LINE3

sub ebx,Zval ; LINE4

add eax,ebx ; LINE5

inc eax ; LINE6

mov Rval,eax ; LINE7

exit

main endp

end main

**Answer:**

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | **Table 2** | |
| LINE1 | **EAX = 0000001Ah** |  | Move the value of Xval (26d) |
| **Xval = 0000001Ah** |  | into register EAX |
|  |  |
| LINE2 | **EAX =FFFFFFE6h** |  | Negate the value of register EAX  (26d become -26d) |
|  |  |  |  |
| LINE3 | **EBX =00000001Eh** |  | Move the value of Yval (30d) into register EBX |
| **Yval =00000001Eh** |  |  |
|  |  |  |
| LINE4 | **EBX = FFFFFFF6h** |  | Subtract the value of register EBX by Zval (40d) |
| **Zval =00000028h** |  |  |
|  |  |  |
| LINE5 | **EAX =FFFFFFDCh** |  | Add the value of register EBX (-10d) to the register EAX(-26d) |
| **EBX =FFFFFFF6h** |  |  |
|  |  |  |
| LINE6 | **EAX =FFFFFFDDh** |  | Increase the register EAX by 1(become -35) |
|  |  |  |  |
| LINE7 | **EAX =FFFFFFDDh** |  | Move the value of register EAX into Rval |
| **Rval =FFFFFFDDh** |  |  |
|  |  |  |

3. Execute the program below. Determine output of the program by inspecting the content of the related registers. Fill in Table 3 with the content of each register or variable on every LINE, in **Hexadecimal** (as per the output). Please complete the comments for every LINE.

**Arithmetic expression: var4 = [(var1 \* var2) + var3] - 1**

include irvine32.inc

.data

var1 DWORD 5

var2 DWORD 10

var3 DWORD 20

var4 DWORD ?

.code

main proc

mov eax, var1 ; LINE1

mul var2 ; LINE2

add eax, var3 ; LINE3

dec eax ; LINE4

exit

main endp

end main

**Answer:**

**Table 3**

|  |
| --- |
|  |
| LINE1 | **EAX = 00000005h** |  | Move the value of var1 (5d) | |
| **var1 = 00000005h** |  | into register EAX | |
|  |  |
| LINE2 | **EAX =00000032h** |  | Multiply the | Register EAX |
| **var2 =0000000Ah** |  | by value of var2(10d) |  |
|  |  |  |  |
| LINE3 | **EAX =00000046h** |  | Add value of | var3(20d) |
| **var3 =00000014h** |  | into register EAX |  |
|  |  |  |  |
| LINE4 | **EAX =00000045h** |  | Decrement value of register EAX by 1 |  |
| **var4 =00000000h** |  |  |  |
|  |  |  |  |

4. Execute the program below. Determine output of the program by inspecting the content of the related registers. Fill in Table 4 with the content of each register or variable on every LINE, in Hexadecimal (as per the output). Please complete the comments for every LINE.

**Arithmetic expression: var4 = (var1 \* 5) / (var2 – 3)**

include irvine32.inc

.data

var1 WORD 40

var2 WORD 10

var4 WORD ?

.code

main proc

mov ax,var1 ; LINE1

mov bx,5 ; LINE2

mul bx ; LINE3

mov bx,var2 ; LINE4

sub bx,3 ; LINE5

div bx ; LINE6

mov var4,ax ; LINE7

exit

main endp

end main

**Answer:**

**Table 4**

|  |  |  |  |
| --- | --- | --- | --- |
| LINE1 | **AX = 0028h** |  | Move the value of var1 (40d) |
| **var1 = 0028h** |  | into register AX |
|  |  |
| LINE2 | **BX =0005h** |  | Move the value of 5d into register BX |
|  |  |  |  |
| LINE3 | **AX =00C8h** |  | Multiply register AX by register BX |
| **BX =0005h** |  |  |
|  |  |  |
| LINE4 | **BX =000Ah** |  | Move the value of var2 (10d) |
| **var2 =000Ah** |  | into register BX |
|  |  |  |
| LINE5 | **BX =0007h** |  | Subtract register BX by 3 |
|  |  |  |  |
|  | **AX =001Ch** |  |  |
| LINE6 | **BX =0007h** |  | Divide register AX by BX, with remainder =DX |
|  | **DX =0004h** |  |  |
| LINE7 | **AX =0001Ch** |  | Move the value of register AX into var4 |
| **var4 =0001Ch** |  |  |
|  |  |  |

**Table 5 – Initial Registers Values**

|  |  |  |  |
| --- | --- | --- | --- |
| **EAX** | **EBX** | **ECX** | **EDX** |
| 10H | 20H | 2H | 0H |

5. Refer to Table 5 above for the following TWO ( 2 ) tasks.

1. Write and execute the instructions below, then fill in the table with the correct value of the registers. Also note the changes.

\*\****Note****: Always start with the initial value of the registers given in Table 5.*

a. MUL BX

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000200H |  | 00000020H | 00000002H | 00000000H |
| b. MUL CX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000020H |  | 00000020H | 00000002H | 00000000H |
| c. MUL AX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000100H |  | 00000020H | 00000002H | 00000000H |
| d. DIV BX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000000H |  | 00000020H | 00000002H | 00000010H |
| e. DIV CX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000008H |  | 00000020H | 00000002H | 00000000H |
| f. DIV AX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000001H |  | 00000020H | 00000002H | 00000000H |

1. Write and execute the instructions below, then fill in the table with the correct value of the registers. Also note the changes.

\*\****Note****: Always start with the initial value of the registers given in Table 5.*

a. MUL EBX

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000200H |  | 00000020H | 00000002H | 00000000H |
| b. MUL ECX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000020H |  | 00000020H | 00000002H | 00000000H |
| c. MUL EAX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000100H |  | 00000020H | 00000002H | 00000000H |
| d. DIV EBX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000000H |  | 00000020H | 00000002H | 00000010H |
| e. DIV ECX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000008H |  | 00000020H | 00000002H | 00000000H |
| f. DIV EAX | |  |  |  |
|  | |  |  |  |
| **EAX** |  | **EBX** | **ECX** | **EDX** |
| 00000001H |  | 00000020H | 00000002H | 00000000H |