

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Programme | : | **BTech. CSE Core** | Semester | : | **Win 2021-22** |
| Course | : | **Microprocessor and Interfacing** | Code | : | **CSE2006** |
| Faculty | : | **Dr. Florence Gnana Poovathy J** | Slot | : | **L15+L16** |
| Name | : | **Hariket Sukesh Kumar Sheth** | Register No. | : | **20BCE1975** |

**Experiment 2:**Calculating the Average of the elements present in Array

**1**

|  |  |  |  |
| --- | --- | --- | --- |
| Date: 19-01-2022 | Exp. 02 | Array Operations |  |

**Average of Array Elements**

**Aim:** To Perform Average of the elements present in the Entered Array

**Tool Used:** Assembler – MASM611

**Algorithm:**

**Step 1:** First of all, mount the c drive using the command: **mount c** **c:\masm611\bin**

**Step 2:** After pressing **enter,** type **c:** and press enter.

**Step 3:** Now give a command, **array.asm** for writing/editing the code and the write the code.

**Step 4:** A pop window appears; there we have to write out code(instructions) following the logic given below.

1. Create an array (DB).
2. Store “DATA” in DS using MOV command.
3. CX is given the value 5H and AX is given the value 0H.
4. Assign SI the beginning address of the array using LEA command.
5. Within the loop the contents of SI are progressively added to AX, and CX is decremented by 1 with each iteration.
6. End the loop using the JNZ command when CX reaches the value 0H.
7. Outside the loop AX is divided by 5H which is moved into BL and divided using DIV.

**Step 5:** Now give a command, **masm array.asm** for running the code. The object file is created.

**Step 6:** Now give a command, **link array.obj** to link the object file to library file present in the bin folder.

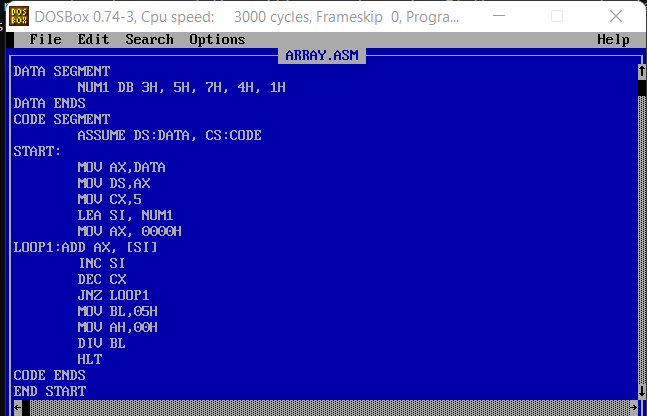
**Step 7:** Press **ENTER** four times.

**Step 8:** Write debug **array.exe**

**2**

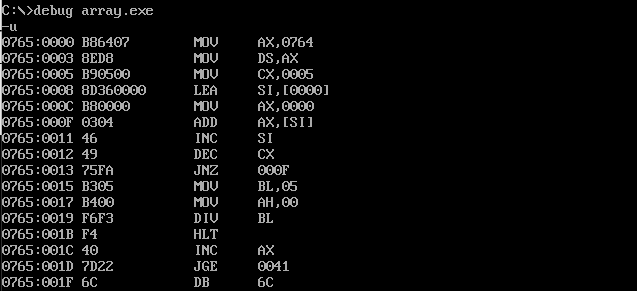
**-u**

**-g** (followed by the **address of HLT or INT** to view the values in registers).

**Program:**

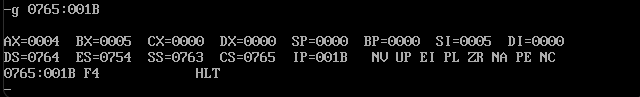
|  |  |
| --- | --- |
| **Sample Input:** | **Sample Output:** |
| Data: 3H, 5H, 7H, 4H, 1H | SUM= 20H Average = 20H/5H = 04H Hence, AX=0004H |

**Register / Memory Contents for I/O:**



**3**

**Output:**



|  |  |  |  |
| --- | --- | --- | --- |
| Date: 19-01-2022 | Exp. 02 | Array Operations |  |

**Sum of Array Elements**

**Aim:** To Perform Sum of the elements present in the Arrays

**Tool Used:** Assembler – MASM611

**Algorithm:**

**Step 1:** First of all, mount the c drive using the command: **mount c** **c:\masm611\bin**

**Step 2:** After pressing **enter,** type **c:** and press enter.

**Step 3:** Now give a command, **sum.asm** for writing/editing the code and the write the code.

**Step 4:** A pop window appears; there we have to write out code(instructions) following the logic given below.

1. Initialize 2 DB arrays NUM1 and NUM2, create a RESULT array too with 4 values of 0
2. Move “DATA” in DS.
3. SI is assigned the beginning address of NUM1, DI is assigned that of NUM2 and BX is assigned that of RESULT using LEA
4. The contents of SI and DI is added and moved into the location represented by BX, via AL
5. Then BX, SI and DL are all incremented within the loop using the LOOP1 statement

**Step 5:** Now give a command, **masm sum.asm** for running the code. The object file is created.

**Step 6:** Now give a command, **link sum.obj** to link the object file to library file present in the bin folder.

**Step 7:** Press **ENTER** four times.

**Step 8:** Write debug **sum.exe**

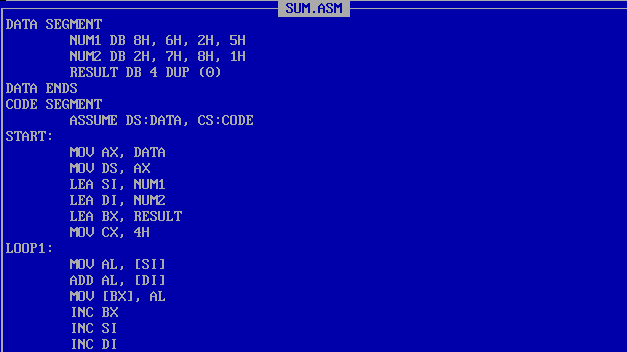
**4**

**-u**

**-g** (followed by the **address of HLT** to view the values in registers).

**-d** (followed by address of Data Segment and index 0 to 7 for viewing array values)

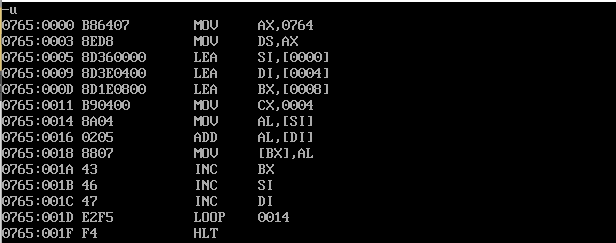
**-d** (followed by address of Data Segment and index 8 to 11 for viewing result)

**Program:**



|  |  |
| --- | --- |
| **Sample Input:** | **Sample Output:** |
| NUM1: 8H, 6H, 2H, 5H NUM2: 2H, 7H, 8H, 1H | Result:  08H+02H = 0AH ; 06H+07H = 0DH 08H+0AH = 0DH ; 05H+01H = 06H |

**Register / Memory Contents for I/O:**



**Output:**

**5**

