

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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 4:**Fibonacci Series, Factorial Number, Permutation and Combination of a Number

**1**

|  |  |  |  |
| --- | --- | --- | --- |
| Date: 09-02-2022 | Exp. 04 | Fibonacci and Factorial Number |  |

**Fibonnaci Numbers**

**Aim:** To print the Fibonacci Series

**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, **fib.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. An array (DB) of 10 values is created, and stored in AX register.
2. SI register is pointed to the start of array. 10 is stored in CX register, and 0 is stored in AL register. The value stored in AL is moved to the start of array, pointed by [SI].
3. AL and SI registers are incremented by 1 each.
4. Again, the value stored in AL is moved to the start of array, pointed by [SI].
5. 0 is stored in AL, and then it heads into the loop L1.
6. Within the loop, the values stored in AL and the starting value of SI are added, and stored in AL register.
7. SI is incremented by 1, and the result of addition is moved from AL to SI register.
8. SI is then decremented by 1, and the value stored in [SI] is moved to AL register, and SI is again incremented.
9. MOVE AH, 4CH is used to interrupt the program, and HLT brings it a halt.

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

**2**

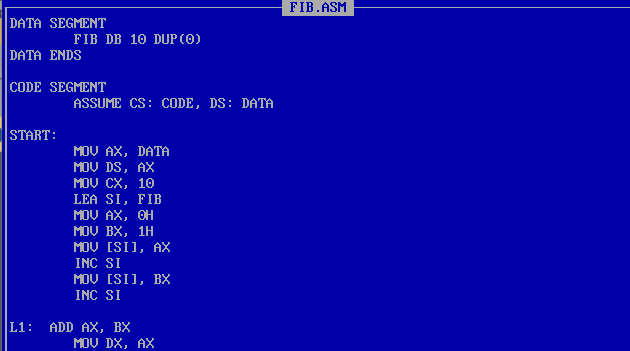
**Step 6:** Now give a command, **link fib.obj** to link the object file to library file present in the bin folder.

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

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

**-u**

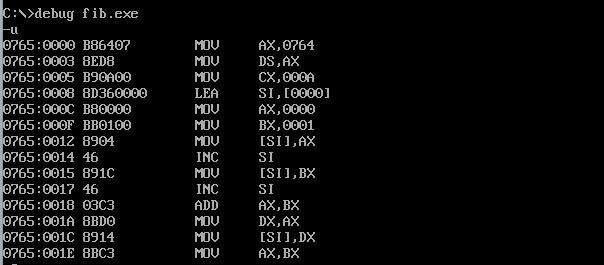
**-g** (followed by the **address of HLT or INT** to view the values in registers).

**Program:**

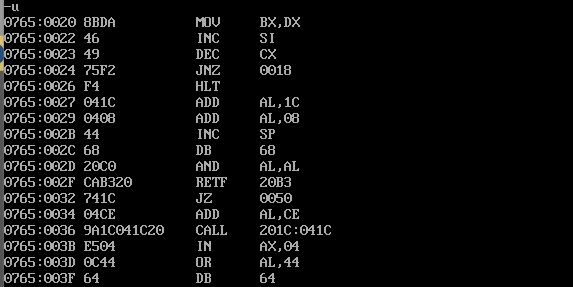


|  |  |
| --- | --- |
| **Sample Input:** | **Sample Output:** |
| AX: 0H, BX: 1H CX: 10 | Fibonacci Series:  0H, 1H, 1H, 2H, 3H, 5H, 8H, 0DH (13), 15H (21), 22H(34) |

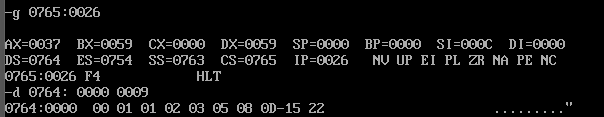
**Register / Memory Contents for I/O:**



**3**



**Output:**



**4**

|  |  |  |  |
| --- | --- | --- | --- |
| Date: 09-02-2022 | Exp. 04 | Fibonacci and Factorial Number |  |

**Factorial Number**

**Aim:** To Perform Factorial of the Number

**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, **fact1975.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. 1H is stored in AX register, and 4H is stored in CX register.
2. Within the loop, the value in CX and AX registers are multiplied and stored in AX, and CX is decremented.
3. The process continues until CX is zero, and then the loop is exited.
4. The final product can be found in AX register.

**Step 5:** Now give a command, **masm fact1975.asm** for running the code. The object file is created.

**Step 6:** Now give a command, **link fact1975.obj** to link the object file to library file present in the bin folder.

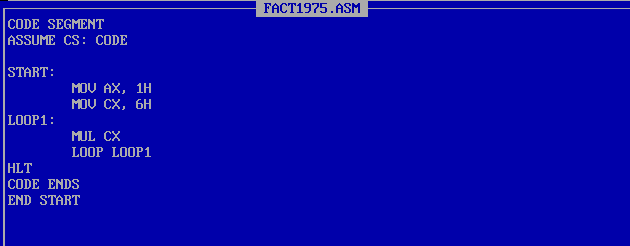
**Step 7:** Press **ENTER** four times.

**Step 8:** Write debug **fact1975.exe**

**-u**

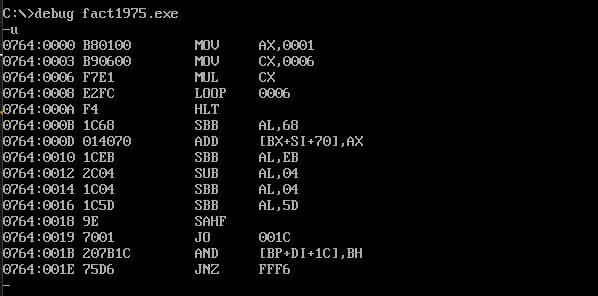
**-g** (followed by the **address of HLT or INT** to view the values in registers).

**5**

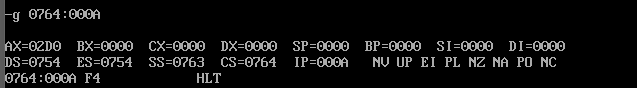
**Program:**

|  |  |
| --- | --- |
| **Sample Input:** | **Sample Output:** |
| Input: 6H | Result AX: 02D0H  = 720 |

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



**Output:**



**6**

|  |  |  |  |
| --- | --- | --- | --- |
| Date: 09-02-2022 | Exp. 05 | Permutation and Combination |  |

**Permutation**

**Aim:** To find the Permutation of the number

**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, **permt.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 CX with N, AX with 1 and BX with r
2. Subtract r from N and store the result in CX
3. Store (N-r)! in AX
4. Load CX with N and move (N-r)! to BX
5. Store N! in AX
6. Divide AX by BX to store the final result in AX

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

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

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

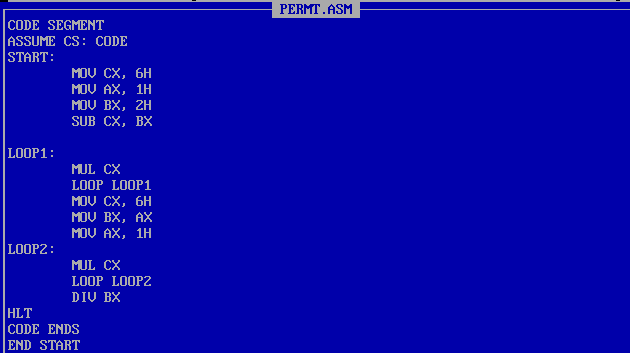
**Step 8:** Write debug **permt.exe**

**-u**

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

**7**

**Program:**

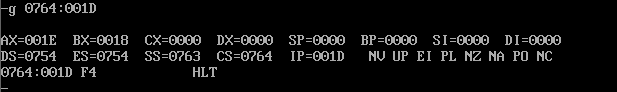
****

|  |  |
| --- | --- |
| **Sample Input:** | **Sample Output:** |
| Input: 6H, 2H 6P2 = 6!/(6-2)! | Result AX: 001EH  = 30 |

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



**Output:**



**8**

|  |  |  |  |
| --- | --- | --- | --- |
| Date: 09-02-2022 | Exp. 05 | Permutation and Combination |  |

**Combination**

**Aim:** To find the Combination of the number

**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, **comb1975.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 CX with N, AX with 1 and BX with r
2. Subtract r from N and store the result in CX
3. Store (N-r)! in AX
4. Load CX with N and move (N-r)! to BX
5. Store N! in AX
6. Divide AX by BX
7. Move r into CX
8. Move the present value of AX which is N!/(N-r)! into BX
9. Store r! in AX
10. 10.Exchange the values of AX and BX.
11. Divide AX by BX to obtain the final result

**Step 5:** Now give a command, **masm comb1975.asm** for running the code. The object file is created.

**9**

**Step 6:** Now give a command, **link comb1975.obj** to link the object file to library file present in the bin folder.

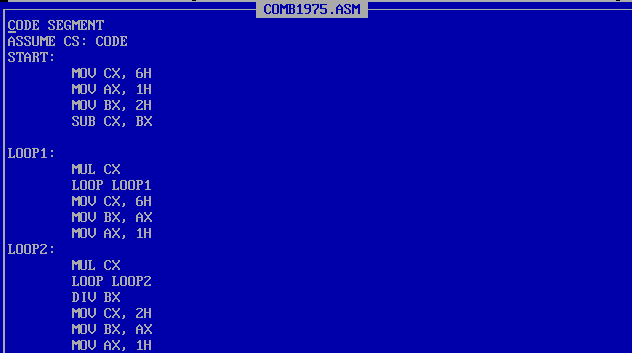
**Step 7:** Press **ENTER** four times.

**Step 8:** Write debug **comb1975.exe**

**-u**

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

**Program:**

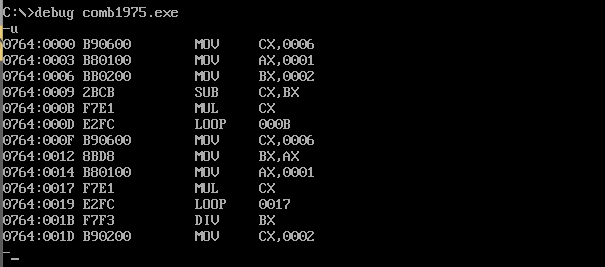
****

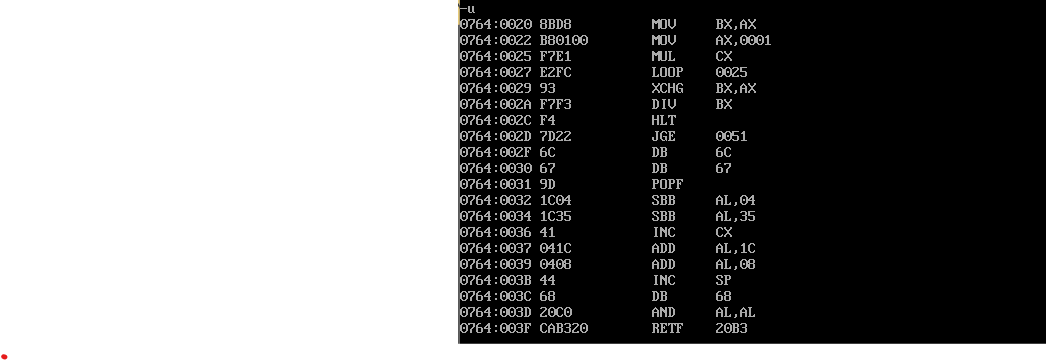
****

|  |  |
| --- | --- |
| **Sample Input:** | **Sample Output:** |
| Input: 6H, 2H 6C2 = 6!/(6-2)!\*2! | Result AX: 000FH  = 15 |

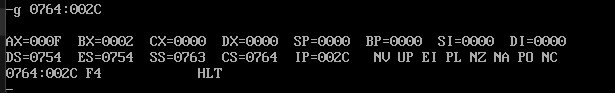
**10**

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





**Output:**



**11**

**11**