



Vidyavardhini's College of Engineering & Technology  
Department of Artificial Intelligence and Data Science (AI&DS)

---

<b>Name:</b>	Shagun Upadhyay
<b>Roll No:</b>	60
<b>Class/Sem:</b>	SE/IV
<b>Experiment No.:</b>	2B
<b>Title:</b>	Program for calculating factorial using assembly language
<b>Date of Performance:</b>	24/01/2024
<b>Date of Submission:</b>	31/01/2024
<b>Marks:</b>	
<b>Sign of Faculty:</b>	



# Vidyavardhini's College of Engineering & Technology

## Department of Artificial Intelligence and Data Science (AI&DS)

---

**Aim:** Program to calculate the Factorial of a number.

**Theory:**

To calculate the factorial of any number, we use MUL instruction. Here, initially, we initialize the first register by value 1. The second register is initialized by the value of the second register. After multiplication, decrement the value of the second register and repeat the multiplying step till the second register value becomes zero. The result is stored in the first register.

**Algorithm:**

1. Start.
2. Set AX=01H, and BX with the value whose factorial we want to find.
3. Multiply AX and BX.
4. Decrement BX=BX-1.
5. Repeat steps 3 and 4 till BX=0.
6. Stop.

Code:

```
ORG 100H

MOV AX, 00H

MOV BX, 04H

MOV CX, 02H

L1: ADD AX, BX

    DEC CX

    JNZ L1

RET
```



The screenshot shows the NoName.com emulator interface. At the top, the title bar reads "emulator: noname.com". Below it is a menu bar with "file", "math", "debug", "view", "external", "virtual devices", "virtual drive", and "help". A toolbar contains icons for "Load", "reload", "step back", "single step", "run", and a "step delay ms: 0" input field.

The main window is divided into several sections. On the left is the "registers" section, which lists 16 registers (AX, BX, CX, DX, CS, IP, SS, SP, BP, SI, DI, DS, ES) with their high (H) and low (L) byte values. The IP register is highlighted with a blue background and shows the value 0154.

In the center is a memory dump section titled "F400:0154". It displays a list of memory addresses and their contents. The address F4154: is highlighted in blue, showing the value CF 207. The dump includes various instructions and data, such as "F4150: FF 255 RES", "F4151: FF 255 RES", "F4152: CD 205 =", "F4153: 20 032 SPA", "F4154: CF 207 =", "F4155: 00 000 NULL", "F4156: 00 000 NULL", "F4157: 00 000 NULL", "F4158: 00 000 NULL", "F4159: 00 000 NULL", "F415A: 00 000 NULL", "F415B: 00 000 NULL", "F415C: 00 000 NULL", "F415D: 00 000 NULL", "F415E: 00 000 NULL", "F415F: 00 000 NULL", "F4160: FF 255 RES", "F4161: FF 255 RES", "F4162: CD 205 =", "F4163: 1A 026 +", "F4164: CF 207 =", "F4165: 00 000 NULL".

On the right is another memory dump section titled "F400:0154". It displays a list of memory addresses and their contents. The address F4154: is highlighted in blue, showing the value CF 207. The dump includes various instructions and data, such as "BIOS DI", "INT 020h", "IRET", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "ADD BH, BH", "DEC BP", "SBB CL, BH", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "ADD BH, BH", "DEC BP", "ADD BH, CL", "ADD [BX + SI], AL", "ADD [BX + SI], AL", "...".

At the bottom of the window is a status bar with labels for "screen", "source", "reset", "aux", "vars", "debug", "stack", and "flags".

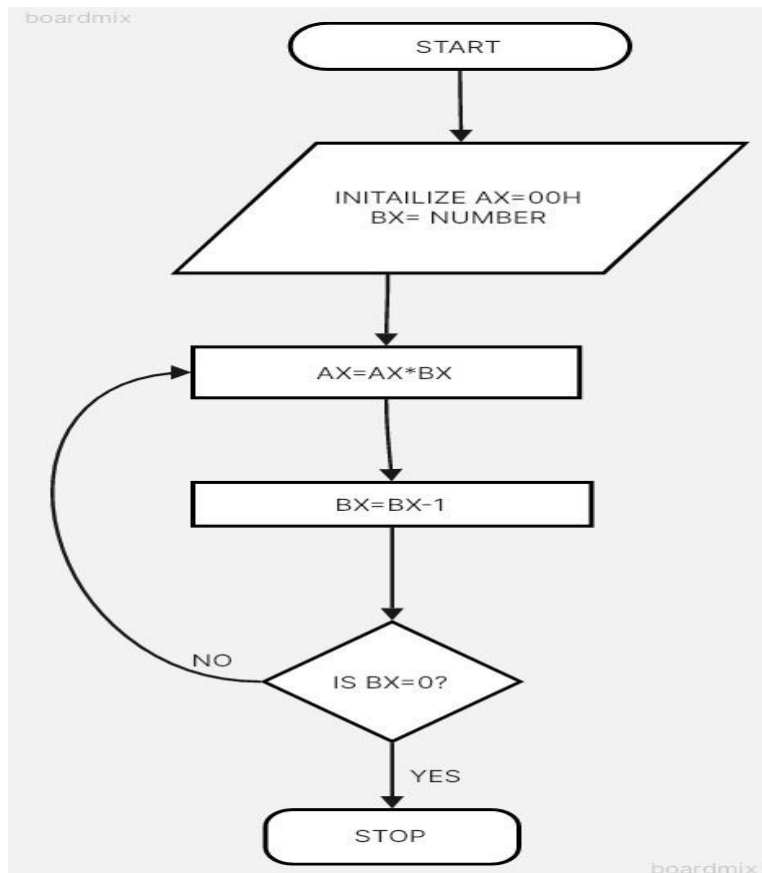


# Vidyavardhini's College of Engineering & Technology

## Department of Artificial Intelligence and Data Science (AI&DS)

---

Flowchart:





Conclusion:

1. Explain shift instructions.
2. Explain rotate instructions.

1)The 8086 can perform two types of Shift operations; the logical shift and the arithmetic shift. There are four shift operations (SHL, SAL, SHR, and SAR).

2)The 8086 can perform two types of rotate operations; the rotate without carry and the rotate through carry. There are four rotate operations (ROL, ROR, RCL, and RCR). ROL shifts each bit of a register to the left. The highest bit is copied into both the Carry flag and into the lowest bit of the register