**SARDAR VALLABHBHAI NATIONAL INSTITUTE OF TECHNOLOGY MIT**

**NAME: KRISHNA PANDEY**

**ADM NO: U20CS110**

**ROLL NO: B110**

ASSIGNMENT-05

1. Write an assembly language program in 8085 to find the factorial of given number using subroutine.

# CODE

;NUMBER IS STORED IN MEMORY AND THE FACTORIAL IS

;STORED IN THE MEMORY LOCATION JUST AFTER THE NUMBER

;STORING VALUES MVI A,05H

STA 3000H

;PROGRAM LXI H,3000H MOV B,M MVI D,01H

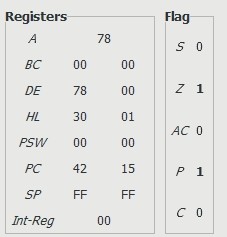
FC: CALL MULT DCR B

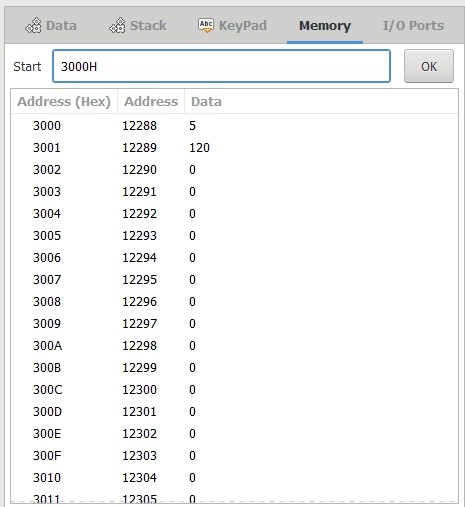
JNZ FC INX H MOV M,D HLT

MULT: MOV C,B MVI A,00H LOOP: ADD D DCR C

JNZ LOOP MOV D,A RET

# OUTPUT





1. Write an assembly language program in 8085 to display Fibonacci series using subroutine.

# CODE

MVI D,00H LXI H,0000H MVI E,08H

MVI A,00H CALL DISP MVI A,01H CALL DISP

LOOP: CALL SERIES CALL DISP

DCR E JZ END

JMP LOOP

SERIES: MOV C,A ADD D

MOV D,C

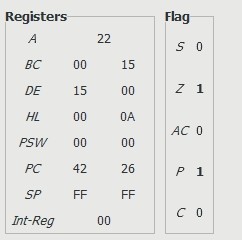
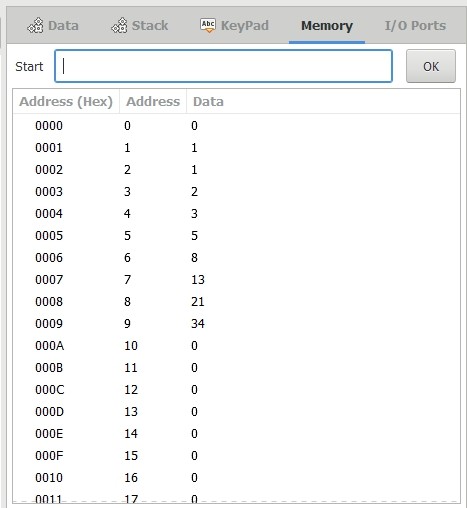
RET

DISP: MOV M,A INX H

RET

END: HLT

# OUTPUT



1. Write an assembly language program in 8085 to multiply two 8 bit numbers using subroutine.

# CODE

;VALUES TO BE MULTIPLIED ARE STORED AT 3000H & 3001H MVI A,05H

STA 3000H MVI A,10H STA 3001H

;OUTPUT IS AT PORT 0 AND REGISTERS B AND C RETAIN THE MULTIPLIED VALUES LXI H,3000H

MOV A,M INX H CMP M JNC CONT MOV B,A MOV E,B MOV C,M JMP CON

CONT: MOV C,A MOV B,M

MOV E,B

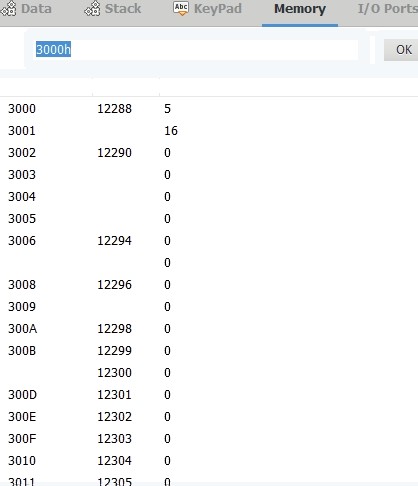
CON: CALL MULT MOV A,D

OUT 00H HLT

MULT: MOV A,E CPI 00H

JZ DISP MOV A,D ADD C MOV D,A DCR E JMP MULT DISP: RET

# OUTPUT



Address (Hex) Address DaLa

12293

3007

12295

300C

Start

DaLa SCack KeyPad Memory Z/O Ports

DK

|  |  |  |
| --- | --- | --- |
| Start  Addi ess (Hex) | Addi ess | Data |
| 00 | 0 | 80 |
| 01 | 1 | 0 |
| 02 | 2 | 0 |
| 03 | 3 | 0 |
| 04 | 4 | 0 |
| 05 | 5 | 0 |
| 06 | 6 | 0 |
| 07 | *7* | 0 |
| 08 | 8 | 0 |
| 09 | 9 | 0 |
| 0A | 10 | 0 |
| 0B | 11 | 0 |
| 0C | 12 | 0 |
| 0D | 13 | 0 |
| 0E | 14 | 0 |
| OF | 15 | 0 |
| 10 | 16 | 0 |
| 11 | 1 7 | L! |

*S* 0

|  |  |  |  |
| --- | --- | --- | --- |
| *A* |  | 50 |  |
| *BC* | 05 |  | 10 |
| 6P | 50 |  | 00 |
| II | 30 |  | 01 |
| PPK | 00 |  | 00 |
| *:fiP El-Rep* | FF | 00 | FF |

*AC* 0

*C* 0