CSE 101 INTRODUCTION to COMPUTER SCIENCE

Homework #02

1)   171044065_Berke_Belgin_Part1.asm

2)   171044065_Berke_Belgin_Part2.asm

3)

B	=	42	=	0100 0010
e	=	65	=	0110 0101
r	=	72	=	0111 0010
k	=	6B	=	0110 1011
e	=	65	=	0110 0101

Berke = 01000010 01100101 01110010 01101011 01100101

4)

```
1- 4AA      +      F12        = 13BC
2- 501      +      4B3        = 9B4
3- 78C      –      1A3        = 5E9
4- 997      –      01E        = 979
5- 10101111 +    11110010 = 1 1010 0001
```

Berke Belgin

171044065

```
                            ;Input Field
        load  R1,5d         ;Enter a value between 1d and 5d (n!)
        move  R2,R1         ;The value to store (n - 1), Equals to (n) at first

                            ;Assignments for Operations
        load  R5,1d         ;+1 number
        load  R6,-1d        ;-1 number
        load  R7,0d         ;0 number
        load  R8,10d        ;10 number
        load  R9,-10d       ;-10 number
        load  RA,100d       ;100 number
        load  RB,-100d      ;-100 number
        load  RC,30h        ;30h number

                            ;Main Function to Operate Factorial Operation
Factorial:  addi  R2,R2,R6  ;R2 = (n - 1) also (R2 - 1)
        move  R0,R7         ;R0 = 0 (for if statement)
        jmpLE R2<=R0,Def    ;If (n - 1) = 0 go to printing phase
        jmp   MulAssign     ;Go for multiplication phase

                            ;Assignments Before Multiplication
MulAssign:  move  R0,R2     ;R0 = n
        load  R3,0d         ;Variable for Multiply Result
        load  R4,0d         ;Variable for Sum Counter
        jmp   Multiply      ;Loop

                            ;Multiplication Operation
Multiply:   jmpEQ R4=R0,MulResult  ;R4 = (n - 1) then go to Multiplication Result
        addi  R3,R3,R1      ;R3 += (n)
        addi  R4,R4,R5      ;R4 += 1
        jmp   Multiply      ;Loop

                            ;Multiplication Result
MulResult:  move  R1,R3     ;Store Result at R1
        jmp   Factorial     ;Go back to factorial the operation

                            ;Define and Print
Def:    move  R0,R1         ;R0 = Factorial Result (for if statements)
        load  RF,52h        ;Write "R"
        load  RF,65h        ;Write "e"
        load  RF,73h        ;Write "s"
        load  RF,75h        ;Write "u"
        load  RF,6Ch        ;Write "l"
        load  RF,74h        ;Write "t"
        load  RF,3Ah        ;Write ":"
        load  RF,20h        ;Write " "
        jmp   In100         ;Go division by 100d phase

                            ;Check if the Result is Greater Than 100d
In100:  load  RC,30h        ;RC = 0 character in ACII
        jmpLE RA<=R0,Dv100  ;if result is greater than 100d Divide by 100d
        move  RF,RC         ;else write 0
        jmp   In10          ;Go division by 10d phase

                            ;Divide by 100d
Dv100:  addi  R0,R0,RB      ;Result -= 100d
        addi  RC,RC,R5      ;ACSII character in 1
```

2 :12   Modified   Insert

```
        load  R0,0d         ;0
        load  R1,1d         ;1
        load  R2,Input      ;Address
        move  R3,R2         ;Address Copy
        load  R4,[R3]       ;Address Value
        load  R5,-1d        ;-1

Count:  jmpEQ R4=R0,Write   ;If the Cell Value = 00 Go Write
        addi  R3,R3,R1      ;Address ++
        load  R4,[R3]       ;R4 = Next Cell Value
        jmp   Count         ;Loop

Write:  move  R0,R2         ;R0 = Start Address
        addi  R3,R3,R5      ;Address --
        load  RF,[R3]       ;Print Cell Value
        jmpEQ R3=R0,Stop    ;If Current Address = Start Address, Stop
        jmp   Write         ;Loop

Stop:   halt                ;Stop

Input:  db    10
        db    "Berke Belgin 171044065"
        db    10
```

24 :1   Insert