Date: 20/11/2022

* Develop an assembly language program to compute nCr using recursive procedure. Assume that 'n' and 'r' are non-negative integers.

```
.MODEL SMALL
.DATA
n dw 4
r dw 2
ncr dw 0

.CODE
mov ax, @DATA
mov DS, AX

mov AX, N
mov bx, r
call ncrpro
call disp
jmp final

ncrpro proc near
        cmp ax, bx      ; r=n
        je resl
        cmp bx, 0       ; r=0
        je resl
        cmp bx, 1       ; r=1
        je resn
        dec ax          ; r=n-1
        cmp bx, ax
        je incr
        push ax
        push bx
        call ncrpro
```

RAJDHANI

Date: / /

```
pop bx
pop ax
dec bx
push ax
push bx
call ncrpro
pop bx
pop ax
ret

resl: inc ncr
ret

incr: inc ncr
resn: add ncr, ax    ; 1+2  3+3=6
ret
ncrpro endp

disp proc near
mov bx, ncr
add bx, 3030h
mov dl, bh
mov ah, 02h
int 21h
mov dl, bl
mov ah, 02h
int 21h
ret
disp endp

final: mov ah, 4ch
int 21h
end
```

RAJDHANI

Date: 20/11/2020

* Write a program to stimulate a Decimal Up-counter to display 00-99.

```
.MODEL SMALL
.CODE
   MOV CL, 00
                ;00
   MOV AH, 00H
   MOV AL, 03H
   INT 10H
BACK: MOV BH, 00H
      MOV DH, 00H
      MOV DL, 00H
      MOV AH, 02H
      INT 10H

   MOV AL, CL
   ADD AL, 00H
   AAM              ;0000
   ADD AX, 3030H    ;3030
   MOV CH, AL
   MOV DL, AH
   MOV AH, 02H
   INT 21H
   MOV DL, CH
   MOV AH, 02H
   INT 21H
   CALL DELAY
   INC CL
   XOR AX, AX
   CMP CL, 1000
   JNE BACK
   JE LAST
```

Date: / /

```
DELAY PROC NEAR
    PUSH AX
    PUSH BX
    PUSH CX
    MOV CX, 00FFH
AG: MOV BX, 0FFH
AG1: NOP
     DEC BX
     JNZ AG1
     DEC CX
     JNZ AG
     POP CX
     POP BX
     POP AX
     RET
DELAY ENDP
LAST: MOV AH, 4CH
      INT 21H
      END
```

Date: 20/ 11 /2020

* Read a pair of input co-ordinates of in BCD and move the cursor to the specified location on screen.

```
.MODEL SMALL
DISP MACRO MSG
  LEA DX, MSG
  MOV AH, 09H
  INT 21H
ENDM
.DATA
  ROW DB 02 DUP(0)
  COL DB 02 DUP(0)
  MSG1 DB 0DH, 0AH, "ENTER X-CO-ORDINATE : $"
  MSG2 DB 0DH, 0AH, "ENTER Y-CO-ORDINATE : $"
  MSG3 db 0dh, 0ah, "cursor displayed at correct
                     coordinates $"

.CODE
    MOV AX, @DATA
    MOV DS, AX ;
    DISP MSG1      ; ROW = 3132   ax = 0102
    MOV SI, OFFSET ROW  ; col = 3233   aad = 0012
    CALL READ
    DISP MSG2      ; 0023
    MOV SI, OFFSET COL
    CALL READ
    MOV SI, OFFSET ROW
     MOV AH, [SI]
      INC SI
     MOV AL, [SI]
     SUB AX, 3030H
      AAD
    MOV DH, AL  ; row
    MOV SI, OFFSET COL
    MOV AH, [SI]
```

RAJDHANI

Date: / /

```
INC SI
MOV AL, [SI]
SUB AX, 3030H
AAD
MOV DL, AL   ;col

MOV AH, 00
MOV AL, 03H
INT 10H
MOV AH, 02H
INT 10H

JMP FINAL

READ PROC NEAR
 MOV CX, 02H
BACK: MOV AH, 01H
      INT 21H
      MOV [SI], AL
      INC SI
      DEC CX
    JNZ BACK
    RET
     READ ENDP
    FINAL: MOV AH, 01H
           INT 21H
           MOV AH, 4CH
           INT 21H
           END
```

RAJDHANI