Problem statement:

Input a string and print longest consecutive subsequence.

Code:

```
.MODEL SMALL
 .STACK 100H

 .DATA
   PROMPT_1  DB  'ENTER A STRING (ALL CAPITAL LETTERS): $'
   PROMPT_2  DB  0DH,0AH,'THE LONGEST CONSECUTIVELY INCREASING STRING IS
: $'
   INVALID   DB  0DH,0AH,'INVALID INPUT. TRY AGAIN : $'

 .CODE
   MAIN PROC
     MOV AX, @DATA
     ; INITIALIZE DS
     MOV DS, AX

     LEA DX, PROMPT_1
     ; LOAD AND DISPLAY THE STRING PROMPT_1
     MOV AH, 9
     INT 21H

     JMP @START

     @TRY_AGAIN: ; JUMP LABEL

     LEA DX, INVALID ; LOAD AND DISPLAY THE STRING INVALID
     MOV AH, 9
     INT 21H

     @START: ; JUMP LABEL

     MOV AH, 1
     INT 21H ; READ A CHARACTER

     CMP AL, 0DH ; COMPARE AL WITH CARRIAGE RETURN
     JE @TRY_AGAIN ; JUMP TO LABEL @TRY_AGAIN IF AL=CR

     CMP AL, 41H  ; COMPARE AL WITH 41H  ('A')
     JB @TRY_AGAIN ; JUMP TO LABEL @TRY_AGAIN IF AL<41H

     CMP AL, 5AH ; COMAPRE AL WITH 5AH  ('Z')
     JA @TRY_AGAIN ; JUMP TO LABEL @TRY_AGAIN IF AL>5AH

     MOV BL, AL;BL=AL  ->
     MOV BH, AL;BH=AL
     MOV DH, AL;DH=AL
     MOV DL, 1 ;DL=1 ->MAX SUBSTRING
     MOV CL, 1 ;CL=1  ->CURRENT MAX SUBSTRING
```

```
      @INPUT:                          ; LOOP LABEL
        INT 21H                        ; READ A CHARACTER

        CMP AL, 0DH                    ; COMPARE AL WITH CR
        JE @END_INPUT                  ; JUMP TO LABEL @END_INPUT IF AL=CR

        CMP AL, 41H                    ; COMPARE AL WITH 41H
        JB @TRY_AGAIN                  ; JUMP TO LABEL @TRY_AGAIN IF AL<41H

        CMP AL, 5AH                    ; COMAPRE AL WITH 5AH
        JA @TRY_AGAIN                  ; JUMP TO LABEL @TRY_AGAIN IF AL>5AH

        INC BL                         ; SET BL=BL+1

        CMP AL, BL                     ; COMPARE AL WITH BL
        JNE @CHECK_AND_REPLACE         ; JUMP TO LABEL @CHECK_AND_REPLACE IF
AL!=BL

        INC CL                         ; SET CL=CL+1
        JMP @INPUT                     ; JUMP TO LABEL @INPUT

        @CHECK_AND_REPLACE:            ; JUMP LABEL

        CMP CL, DL                     ; COMPARE CL WITH DL
        JLE @SKIP_UPDATION_1           ; JUMP TO LABEL @SKIP_UPDATION_1 IF
CL<=DL

        MOV DH, BH                     ; SET DH=BH
        MOV DL, CL                     ; SET DL=CL

        @SKIP_UPDATION_1:              ; JUMP LABEL

        MOV BH, AL                     ; SET BH=AL
        MOV BL, AL                     ; SET BL=AL
        MOV CL, 1                      ; SET CL=1
      JMP @INPUT                       ; JUMP TO LABEL @INPUT

      @END_INPUT:                      ; JUMP LABEL

      CMP CL, DL                       ; COMPARE CL WITH DL
      JLE @SKIP_UPDATION_2             ; JUMP TO LABEL @SKIP_UPDATION_2 IF
CL<=DL

      MOV DH, BH                       ; SET DH=BH
      MOV DL, CL                       ; SET DL=CL

      @SKIP_UPDATION_2:                ; JUMP LABEL

      MOV BX, DX                       ; SET BX=DX

      LEA DX, PROMPT_2                 ; LOAD AND DISPLAY THE STRING PROMPT_2
      MOV AH, 9
      INT 21H
```

```
    XOR CX, CX                      ; CLEAR CX
    MOV CL, BL                      ; SET CL=BL

    MOV DL, BH                      ; SET DL=BH
    MOV AH, 2                       ; SET OUTPUT FUNCTION

    @OUTPUT:                        ; LOOP LABEL
      INT 21H                       ; PRINT A CHARACTER
      INC DL                        ; SET DL=DL+1
    LOOP @OUTPUT                    ; JUMP TO LABEL @OUTPUT IF CX!=0

    EXIT:
    MOV AH, 4CH                     ; RETURN CONTROL TO DOS
    INT 21H
  MAIN ENDP
END MAIN
```

Output: