Experiment Name :Find the maximum length of longest consecutive sequence and the longest  
consecutive sequence in a string.

Theory :The objective of this program is to find the maximum length of the letter which is ordered  
increasingly and no letter is missing between the letters. For example, if a given string  
is “aertdefgabghj” then the maximum length will be – 4 (defg) as they are the longest consecutive substring of that main string.

**Code:**

.MODEL SMALL

.STACK 100H

.DATA

A DB 0

C DB 0

MSG1 DB 'Enter a string : $'

MSG2 DB 0DH,0AH, 'longest consecutive sequence : $'

MSG3 DB 0DH,0AH, 'Maximum length is : $'

.CODE

MAIN PROC

MOV AX,@DATA

MOV DS,AX

MOV AH,9

LEA DX,MSG1

INT 21H

START:

MOV AH,1

INT 21H

INC A

JE LEVEL\_1

MOV CL,1

MOV BL,AL

MOV DH,AL

INPUTS:

INT 21H

INC A

CMP AL,0DH

JE LEVEL\_1

INC BL

CMP BL,AL

JNE INI

INC CL

JMP INPUTS

INI:

CMP CH,CL

JL UPDATE

MOV CL,1

MOV BL,AL

MOV DH,AL

JMP INPUTS

UPDATE:

MOV CH,CL

MOV BH,DH

MOV CL,1

MOV BL,AL

MOV DH,AL

JMP INPUTS

LEVEL\_1:

CMP CH,CL

JL UPDATE2

JMP LEVEL\_2

UPDATE2:

MOV CH,CL

MOV BH,DH

JMP LEVEL\_2

LEVEL\_2:

MOV AH,9

LEA DX,MSG2

INT 21H

MOV AH,2

MOV DL,BH

MOV CL,CH

OUTPUT:

CMP CH,0

JE FINISH

ADD C,1

DEC CH

INT 21H

INC DL

JMP OUTPUT

FINISH:

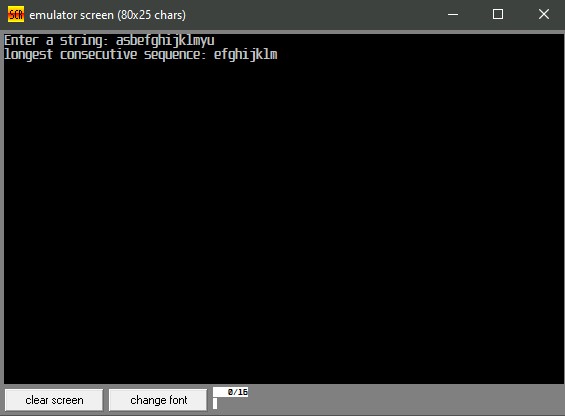
MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

**Output:**



**Discussion:**

In this above program, it will find the longest consecutive sequence of a given string and will count the length of that longest consecutive sequence. To implement the program many functions are used and conditional operator like JE ,JNE ,JNGE etc. is used. And for storing message string double (DB) type MSG1,MSG2 variables were declared and used in the program.

Experiment Name :

Write a program that identifies a hexadecimal number, EVEN or ODD using Shift.

**Theory:**

The objective of this program is to identify even or odd a hexadecimal number and it needs to be done using the left shift method of Assembly language. A hexadecimal number is basically holds 0-9 and A-F. Left Shift in Assembly means the bitwise shift towards left of a binary stream.

**Code:**

.MODEL SMALL

.STACK 100H

.DATA

M1 DB 'TYPE A HEX NUMBER: $'

M2 DB 'THE NUMBER IS EVEN$'

M3 DB 'THE NUMBER IS ODD$'

.CODE

MAIN PROC

MOV AX, @DATA

MOV DS, AX

XOR BX,BX

MOV AH, 9

LEA DX, M1

INT 21H

INPUT\_LOOP:

MOV AH, 1

INT 21H

CMP AL, 0DH

JE CHECK

CMP AL, 39H

JG LETTER

AND AL, 0FH

JMP SHIFT

LETTER:

SUB AL, 37H

SHIFT:

SHL BX, 4

OR BL, AL

JMP INPUT\_LOOP

CHECK:

SHR BX,1

JNC EVEN:

MOV AH, 2

MOV DL, 0DH

INT 21H

MOV DL, 0AH

INT 21H

MOV AH, 9

LEA DX, M3

INT 21H

JMP L

EVEN:

MOV AH, 2

MOV DL, 0DH

INT 21H

MOV DL, 0AH

INT 21H

MOV AH, 9

LEA DX, M2

INT 21H

L:

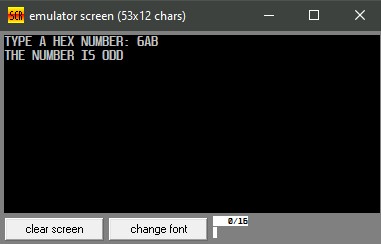
MOV AH, 4CH

INT 21H

MAIN ENDP

END MAIN

**Output:**



**Discussion:**

In the above assembly code, a hexadecimal number was taken as input using loop. After that, That very hex number was right shifted once. Then the Carry Flag (CF) was checked. If CF= 0 then the number is even, otherwise it is odd. That is how the program was done.