**PROGRAMS**

1. **Write an ALP to Display Fibonacci series**

|  |  |
| --- | --- |
| **ASSUME CS:CODE, DS:DATA**  **DATA SEGMENT**  **COUNT DB 0AH**  **FIB DB 10 DUP(?)**  **DATA ENDS**  **CODE SEGMENT**  **START:**  **MOV AX, DATA**  **MOV DS, AX**  **MOV CL,COUNT**  **LEA SI,LIST**  **MOV AL,0H**  **MOV [SI],AL**  **MOV BL,01H**  **GO: INC SI**  **ADD BL,AL**  **MOV [SI],BL**  **MOV AL,[SI-1]**  **DEC CL**  **JNZ GO**  **INT 03H**  **CODE ENDS**  **END START** | **Result**  **Data Segment** |

1. **Write an ALP to move a string of data bytes form one location to another**

|  |  |
| --- | --- |
| **ASSUME CS:CODE, DS:DATA**  **DATA SEGMENT**  **STR1 DB ‘CSE’**  **DATA ENDS**  **EXTRA SEGMENT**  **STR2 DB ‘00H’**  **EXTRA ENDS**  **CODE SEGMENT**  **START:**  **MOV AX, DATA**  **MOV DS, AX**  **MOV AX, EXTRA**  **MOV ES, AX**  **LEA SI, STR1**  **LEA DI, STR2**  **MOV CL,03H**  **CLD**  **REP MOVSB**  **INT 03H**  **CODE ENDS**  **END START** | **Result**  **Data Segment**    **Extra Segment** |

1. **Write an ALP to concatenate two strings**

|  |  |
| --- | --- |
| **ASSUME CS:CODE, DS:DATA**  **DATA SEGMENT**  **STR1 DB ‘CSE’**  **STR2 DB ‘SNIST’**  **STR3 DB 00H**  **DATA ENDS**  **CODE SEGMENT**  **START:**  **MOV AX,DATA**  **MOV DS,AX**  **MOV AX,0H**  **MOV CL,08H**  **LEA SI,STR1**  **LEA DI,STR3**  **GO:**  **MOV AL,[SI]**  **MOV [DI],AL**  **INC SI**  **INC DI**  **DEC CL**  **JNZ GO**  **INT 03H**  **CODE ENDS**  **END START** | **Result**  **Data Segment** |

1. **Write an ALP to reverse a given string**

|  |  |
| --- | --- |
| **ASSUME CS:CODE, DS:DATA,**  **DATA SEGMENT**  **STR1 DB ‘CSE’**  **DATA ENDS**  **CODE SEGMENT**  **START:**  **MOV AX, DATA**  **MOV DS, AX**  **MOV AX,0H**  **MOV CL,03H**  **LEA SI,STR1**  **LEA DI,STR+6**  **GO:**  **MOV AL,[SI]**  **MOV [DI],AL**  **INC SI**  **DEC DI**  **DEC CL**  **JNZ GO**  **INT 03H**  **CODE ENDS**  **END START** | **Result**  **Data Segment** |