String

No Jump statement of any type should be used in following programs. Do not use [,].

```
.model small
.data
 a db "pqadrtftyg$"
.code
  Mov AX,Dgroup
  Mov ES,AX
  Mov DS,AX
  Lea DI,a
  Add DI,5
  Mov AL,'x'
  CLD
  Stosb
  Stosb
  Lea DX,a
  Mov AH,9
  Int 33
  Mov AH,76
  Int 33
  end
```

Stosb $\equiv$ ES:[DI]$\leftarrow$AL   DI=DI+1

Hence the given program outputs pqadrxxtyg

When CLD is replaced by STD then output pqadxxftyg

When CLD is used then Stosb increments DI.

When STD is used then Stosb decrements DI.

Rep Stosb $\equiv$ The instruction Stosb is executed CX times.

After every execution (of stosb) CX is decremented and DI is incremented (decremented in case of std). When both Stosb are removed and Mov CX,3 and rep Stosb is put then o/p pqadrxxxyg   while(CX$\neq$0){CX-- Stosb DI++ }

In general   **Rep T** $\equiv$ The instruction T is executed CX times.

1. Read a number (n, less than10) make first n letters of string as 'x'. Let n=4 then xxxxrtftyg   n=6xxxxxxftyg

2. Do above problem using STD.

3. Make n^{th} letter as x. n=4 pqadxtftyg. n=8 pqadrtftxg

4. Read two single digit numbers p and q (p<q). Replace all letters between locations p and q by x. Input 25   output pqxxxxftyg   Input 03 output xxxdrtftyg

    Lodsb $\equiv$ AL $\leftarrow$ DS:[SI]   SI=SI+1   Rep Lodsb is similar

5. Output 4th letter without using add. In string 'a' it is 'r'

6. Output 4th and 3rd letter. [Hint: first cld then Std] Here rd. No add/subtract. Lea once

7. Output 7th letter then 3rd letter then 9th letter. tdg[No add/sub, use cld/std, lea once]

8. Read two single digit numbers x and y. Output x+y (let it is less than 10). No add subtract is to be used. [Hint: take string 'a' as "0123456789$"] [Sub al,48 permitted]

9. Output x$-$y assume x is greater than y. [Sub al,48 permitted/not]

10. Find 7th letter. Let it be digit 'k'. Output $(7-k)^{th}$ letter. [no add/sub] [Let a="318732452794$" o/p 8.] [Let a="3187612387345" o/p 6.] [Sub al,48 permitted]

11. Find 7th letter. Let it be 'k'. Let 'm' be $(7-k)^{th}$ letter. Output $(7-k+m)^{th}$ Letter. (9,3)

12. Do above 4 problems without using sub al,48.

13. Replace 4th, 5th, 6th, 7th letters by 2nd letter.

```
.model small
.data
 a db "tttfghtwer$"
 b db "afertdfgyq$"
.code
  Mov AX,Dgroup
  Mov ES,AX
  Mov DS,AX
  Lea DI,a
  Mov AL,'t'
  Mov CX,12
  CLD
  Repe Scasb
  Mov AH,2
  Mov DX,CX
  Add DL,48
  Int 33
  Mov AH,76
  Int 33
End
```

Scasb ≡ Cmp AL,ES:[DI] (flag setting) DI=DI+1

Repe Scasb ≡ Scasb is executed at most CX times. After every execution CX is decremented. If any comparison leads to inequality then further execution of Scasb does not take place. Output of the given program is (12-4=8) because the scasb is executed 4 times. Now [DI] points at 'g'. When CX is large then DI points at the next letter after first non-AL.

if $(CX \neq 0)$ do$\{CX--\}$while $(AL=ES:[DI++])$ and $(CX \neq 0)$

Repne Scasb: When a comparison leads to equality then no further execution. If Repne is used on string b then output is 7. In this program if Repne scasb is used after Repe scasb then CX will be (8-3=5). Next letter after first AL.

14. Letter immediately after first 'g'. ['h' in a 'y' in b][Lodsb]

15. Letter immediately before 'g' ['f' in a and b] [Std]

16. Letter immediately after second 'g'. in hjgyuwgrtgd o/p r

17. Letter immediately after first group of 'g'. In tuggghj o/p h.

18. Letter before last 'e' before first 'k'. In yedgetykqejk o/p 'g'. In aretydker o/p 'r'[std]

19. Replace first 'g' by 'z'. [in example tttfzhtwer afertdfzyq]

20. Replace 3 letters before first 'g' by 'x'. [in example txxxghtwer aferxxxgyq]

21. Replace last letter by x. in example [tttfghtwex afertdfgyx] [Hint: Al='$']

22. Output the number of leading t's (CX(initial)-final-1)=3 and 0 respectively.

23. Replace leading t's by x. in example [xxxfghtwer afertdfgyq]

24. Output the number of trailing t's. In string "ghwertttt" o/p 4.

25. Replace trailing t's by x. In string "ghwertttt" o/p ghwerxxxx.

26. Given a string of digits. Add 1 to it. [Replace trailing 9's by 0, Increment digit before it] [Example: Let string a="231999$". It is modified to 232000.]

27. Replace first group of g's h. qwegggyusdgggjkl to qwehhhyusdgggjkl.

28. Replace all letters between first two k's by last letter. rakwerktyu$ to rakuuuktyu$

29. Find location of 2nd 'f'. In string 'b' 6. [Use Repne twice]

30. Find location of last 'h' before first 't'. Let string is weherhmhxytasdhtwer then output 7.

31. Let a string is words is given. Let string be ram prasad kumar hari gopal. Output first word. ram

32. Second word.

33. Last word. gopal

34. Length of last word. (5)

35. Replace all letters of first word by 't'. The output is ttt Prasad kumar hari gopal.

36. Replace last 3 letters of second word by 't'. ram prattt kumar hari gopal.

37. Copy the string in another string by deleting second word.

Do above problems assuming that there may be multiple blanks between words.
There may be blanks before the first word and after the last word.
Example: " ram   Prasad   kumar   hari gopal"

```
----------------------------------------------------------------
.model small
.data
  a db "apqawer$"
  b db "xyzwsdf$"
.code
  Mov AX,Dgroup
  Mov ES,AX
  Mov DS,AX
  Lea DI,a
  Lea SI,b
  Add SI,2
  Movsb
  Lea DX,a
  Mov AH,9
  Int 33
  Mov AH,76
  Int 33
End
```

Movsb ≡ ES:[DI]←DS:[SI] DI=DI++;SI++ (STD DI--, SI--)

Output of the given program is zpqawer

38. Replace first 4 letters of 'a' by those of 'b'. New 'a' xyzwwer.

39. Do above using std.

40. Replace last letter of 'a' by first letter of 'b'. apqawex

41. Replace last 4 letters of string 'a' by first 4 letters of string 'b' using std. apqxyzw.

42. Replace first letter of 'a' by last letter of 'b'. (hard)

Cmpsb ≡ Cmp DS:[SI],ES:[DI]   DI=DI++ SI++ flag setting.

43. Compare first letters of string 'a' and 'b'. If 'a' has bigger then output 'a' else output 'b'. (Jmp, JG permitted)

44. Find the first letter of string 'a' and 'b' which is same. [Hint: repne]. If 'a'=wertyeio and 'b'=asdtwes then output is 't'

45. Find second letter which is same. In above case 'e'.

46. Find the first letter of string 'a' which is differing with corresponding letter of string 'b'. Let a=ramwert b=ramwkjh Output 'e'. [Hint: Repe]

47. Output lexicographically bigger string. Let strings are ratyu and ratuiw then ratyu is output. [Find first letter which is different]

48. Find place of first match in string 'a' and 'b'. Replace string 'b' before that by corresponding part of string 'a'. Let a=mahesh$ b=kuhwrty$ the new b=mahwrty$ [ramtyut,jhukyr,ramtyr]

49. Find place of first match in string 'a' and 'b'. Replace string 'b' after that by corresponding part of string 'a'. Let a=mahesh$ b=kuhwrty$ the new b=kuhesh$ [ramtyut,jhukyr,jkukyut]

In the following programs the use of one jump statement is permitted. Let a string is of words is given. Let string be ram prasad kumar hari gopal.

50. Output short form. rpkhg

51. Output kth word. k=3 outputs kumar

52. Output word wise reverse. gopal hari kumar prasad ram

53. Output every word in separate line.

54. How many leading words have same length? Let string be atr ram pqr tyu ghij kl jki then output is 4. Here word lengths are 3333423.