Function

```
import java.io.*;
import java.lang.*;
class f
{  public static String rdstring() throws Exception
   {  DataInputStream o=new DataInputStream(System.in);
      String a=o.readLine();
      return a;
   }
   public static int second(String a,char e)
   {  String b;int p,q;
      p=a.indexOf(e);
      b=a.substring(p+1);
      q=b.indexOf(e);
      return(p+q+1);
   }
   public static String del(String a,int k)
   {  String b,c;
      b=a.substring(0,k)+a.substring(k+1);
      return b;
   }
   public static String delsec(String a,char e)
   {  int p;
      p=second(a,e);
      return del(a,p);
   }
}
class kapil
{  public static void main( String args[])throws Exception
   { String a,k;int g,m;
     a=f.rdstring();g=f.second(a,’x’);System.out.println(g);
   }
}
```

When we execute "Java kapil" then output is the location of second 'x' in the input string. If input is pwerxtyxasxwe then output is 7.

```
class ram
{  public static void main( String args[])throws Exception
   { String a,k;int m;
     a=f.rdstring();m=Integer.parseInt(f.rdstring());
     k=f.del(a,m);System.out.println(k);
   }
}
```

When we execute "Java ram" then input is a string and a number(m). The m^th letter from the string is deleted. If input string is elephant and number is 5 then output is elephnt.

```
class gopal
{  public static void main( String args[]) throws Exception
   { String a,b,k;char c;
     a=f.rdstring();b=f.rdstring();c=b.charAt(0);
     k=f.delsec(a,c);System.out.println(k);
   }
}
```

When we execute "Java gopal" then input is a string and a letter. The second occurrence of the letter is deleted from the string. If input string is pweyqtyuioyer and letter is y then output is pweyqtuioyer.

1. Define function String delkth(String a, int k). It returns a string after deleting k^th letter. [call of delkth("ramharikumar",4) will return "ramhrikumar".
2. Define function String delfirst(String a). It deletes first 'x' in the string. Do not use + in this function. Do not use loop. [Hint: use delkth function.]
3. Define function String delfirsttwo(String a). It deletes first two x's in the string. Do not use indexOf, loop or + in this function. [Hint: use delfirst function twice]
4. Define function int secloc(String a). It finds the location of second x in the string. Use + only once. Do not use second. [call of secloc(ramxharixkapil) will return 8]

5. Define function int firstloc(String a). It finds the location of first x in the string. Do not use indexOf. [Hint: Join ‘x’ in the beginning of first string]

In following functions at most one loop should be used.

6. Define function String delfirstk(String a, int k). It deletes first k x’s in the string. Do not use indexOf or + in this function. [Hint: use delfirst function in a loop]
7. Define function String delall(String a). It deletes all x’s in the string. Do not use + and substring in this function [Hint: use delfirst function in while loop]
8. Define function int count(String a). It returns the number of x’s in the string a. count(“arxtyxxexr”) will return 4.
9. Define function int loc(String a, int k). It finds location of k^{th} x in the string. Do not use loop. [Hint: find location of first ‘x’ after deleting (k-1) x’s.]
10. Define function String aft(String a, char b). It finds first letter after every b. aft(“amwmthmc”,’m’) will return “wtc”. Assume that last letter is not b.
11. Define function String mr(String a). It returns “yes” if first letter is present more than once. mr(“hari”) returns “no”. mr(“abhay”) returns “yes”. Do not use loop.
12. Define function String more(String a). It returns those words of string ‘a’ whose first letter is present in the same word more than once. more(“gtygwe kiuip eleet klimp tyutty gtgy”) will return “gtygwe eleet tyutty gtgy”. [Do not use charAt. Use mr]
13. Define function int cnt(String a, String b). Its returns the number of letters of string a present in string b. cnt(“acgcklm”,”rdgclp”) will return 4. Here c, g, c and l are present. a, k and m are not present.
14. Define function int length(String a). It returns the length of the string ‘a’.
15. Define function String subset(String a, String b). It returns “yes” if every letter of first string is present in second string. subset(“hari”,”iphyearus”) returns “yes”. subset(“klcm”,”kldm”) returns “no”. Do not use loop. [Hint: use cnt and length]
16. Define above function subset using loop. Do not use cnt and loop.
17. Define function, which returns the string of first letter of every word. first(“ram hari om kapil ravi”) will return “rhokr”.
18. Define function String present(String a, String b). It returns string of those words of string ‘a’ whose every letter is present in ‘b’. present(“ram is a good boy”,”tsoaigkdr”) will return “is a good”. [Hint: use subset]
19. Define function String presentfirst(String a, String b). It returns string of first letter of those words of string ‘a’ whose every letter is present in ‘b’. presentfirst(“ram is a good boy”,”tsoaigkdr”) will return “iag”. Do not use loop. Use present and first.
20. Define function String prfirst(String a, String b). It returns string of first letter of those words of string ‘a’ whose every letter after first letter is present in ‘b’. prfirst(“ram is good boy”,”asodg”) will return “ig”.
21. Define function String abcfirst(String a). It returns string of those words of string ‘a’ whose every letter is first letter of some word. abcfirst(“yes ret mrts toy srtk om”) will return “mrts toy om”. No loop [Hint: use first and present]
22. Define function, which returns string of those words, whose first letter is capital. Capital(“kapil Ram Gopal sani”) will return “Ram Gopal”.
23. Define reverse. Reverse(”hari”) returns “irah”.
24. Define rev. It reverses a string in a given range. rev(“hariprasad”,4,8) returns harisarpad.
25. Define function to perform word wise reverse. Rv1(“ram is bad”) returns “bad is ram”.
26. Define function to perform within word reverse. Rv2(“ram is bad”) returns “mar si dab”.
27. Do the problem of word wise reverse using rev and Rv1. No loop.
28. Do the problem of within word reverse using rev and Rv2. No loop.
29. String kth_letter(String a, int k) returns the string of k^{th} letter of every word. Assume that every letter has length at least ‘k’. kth_letter(“ram hari gopal”,2) returns mrp.
30. Define shuffle(“abc pqr ghk tih jkl”) returns “apgtj bqhik crkhl”. Assume all words have same length. [Hint: use above function]

Define each of the above functions without using any other function. More than one loop may be used.