1. Find first last repeating character in a given string.

```
package JAVA;

public class lastrepet
{

	public static void main(String[] args)
	{
		String str ="language";
		int l=str.length();
		String str1 ="";

		for(int i =0;i<l;i++)
		{

			for(int j=0;j<l-1;j++)
			{
				if(str.charAt(i)==str.charAt(j))
				{
					str1=str1+str.charAt(i);
				}
			}
		}
		int l1=str1.length();
		System.out.println("The last repeated charechter in the string is:" +
str1.charAt(l1-1));
	}
}
```

2. Reverse words in a given string without using predefined functions.

```
package JAVA;

public class reverseword
{

	public static void main(String[] args)
	{
		String str="i love programming very much";
		String res="";
		int l=str.length();
		String words[] = str.split("\\s");
		for (int i = 0; i < words.length; i++) {
	if (i == words.length - 1)
	  res = words[i] + res;
	else
	  res = " " + words[i] + res;
      }


	   System.out.print("Reversed string : " + res);
	}
}
```

3. Find count of distinct subsequences in a string.

```
package JAVA;

public class cousub
{

	public static void main(String[] args)
	{
		String str ="var";
		int n = str.length();
	    int[] dp = new int[n + 1];
	    dp[0] = 1;
	    int[] lastOcc = new int[256];
	    for (int i = 0; i < 256; i++)
	    {
		lastOcc[i] = -1;
	    }

	    for (int i = 1; i <= n; i++)
	    {
		char cc = str.charAt(i - 1);
		dp[i] = 2 * dp[i - 1];

		if (lastOcc[cc] != -1) {
		    dp[i] -= dp[lastOcc[cc]];
		}

		lastOcc[cc] = i - 1;
	    }

		System.out.println(dp[n]);
	  }
```