

Permutations.java

Below is the syntax highlighted version of `Permutations.java` from §2.3 Recursion.

```

/*****
 *  Compilation:  javac Permutations.java
 *  Execution:    java Permutations n
 *
 *  Enumerates all permutations on n elements.
 *  Two different approaches are included.
 *
 *  % java Permutations 3
 *  abc
 *  acb
 *  bac
 *  bca
 *  cab
 *  cba
 *
 *****/

public class Permutations {

    // print n! permutation of the characters of the string s (in order)
    public static void perm1(String s) { perm1("", s); }
    private static void perm1(String prefix, String s) {
        int n = s.length();
        if (n == 0) StdOut.println(prefix);
        else {
            for (int i = 0; i < n; i++)
                perm1(prefix + s.charAt(i), s.substring(0, i) + s.substring(i+1, n));
        }
    }

    // print n! permutation of the elements of array a (not in order)
    public static void perm2(String s) {
        int n = s.length();
        char[] a = new char[n];
        for (int i = 0; i < n; i++)
            a[i] = s.charAt(i);
        perm2(a, n);
    }

    private static void perm2(char[] a, int n) {
        if (n == 1) {
            StdOut.println(new String(a));
            return;
        }
        for (int i = 0; i < n; i++) {
            swap(a, i, n-1);
            perm2(a, n-1);
            swap(a, i, n-1);
        }
    }

    // swap the characters at indices i and j
    private static void swap(char[] a, int i, int j) {
        char c = a[i];
        a[i] = a[j];
        a[j] = c;
    }
}
```

```
public static void main(String[] args) {  
    int n = Integer.parseInt(args[0]);  
    String alphabet = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ";  
    String elements = alphabet.substring(0, n);  
    perm1(elements);  
    StdOut.println();  
    perm2(elements);  
}  
}
```

*Copyright © 2000–2017, Robert Sedgewick and Kevin Wayne.
Last updated: Fri Oct 20 14:12:12 EDT 2017.*