

Write a C program to print all permutations of a given string

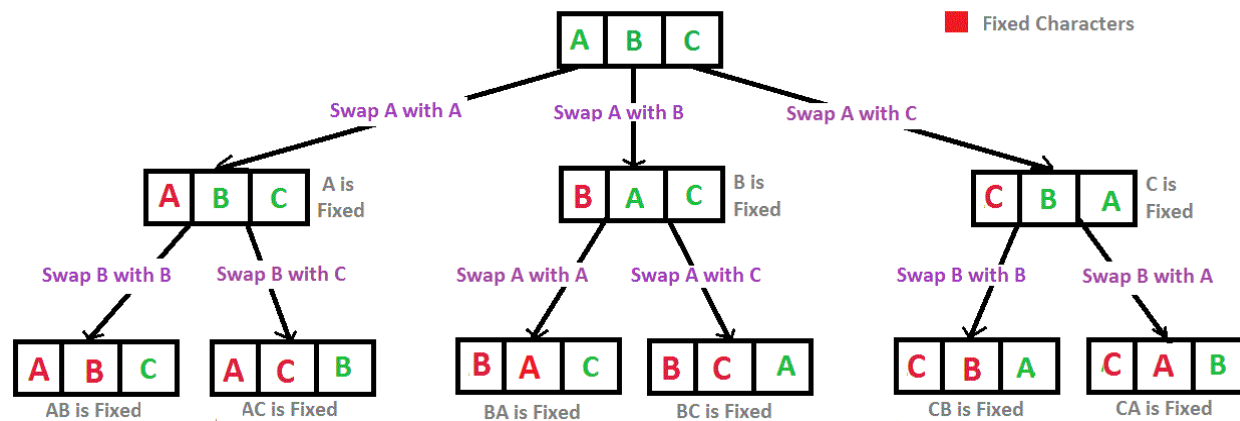
A permutation, also called an “arrangement number” or “order,” is a rearrangement of the elements of an ordered list S into a one-to-one correspondence with S itself. A string of length n has $n!$ permutation.

Source: Mathworld(<http://mathworld.wolfram.com/Permutation.html>)

Below are the permutations of string ABC.

ABC, ACB, BAC, BCA, CAB, CBA

Here is a solution using backtracking.



Recursion Tree for Permutations of String "ABC"

```
# include <stdio.h>
```

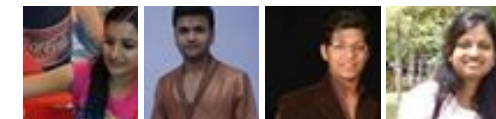
Google™ Custom Search



GeeksforGeeks



53,525 people like GeeksforGeeks.



Interview Experiences

Advanced Data Structures

Dynamic Programming

Greedy Algorithms

Backtracking

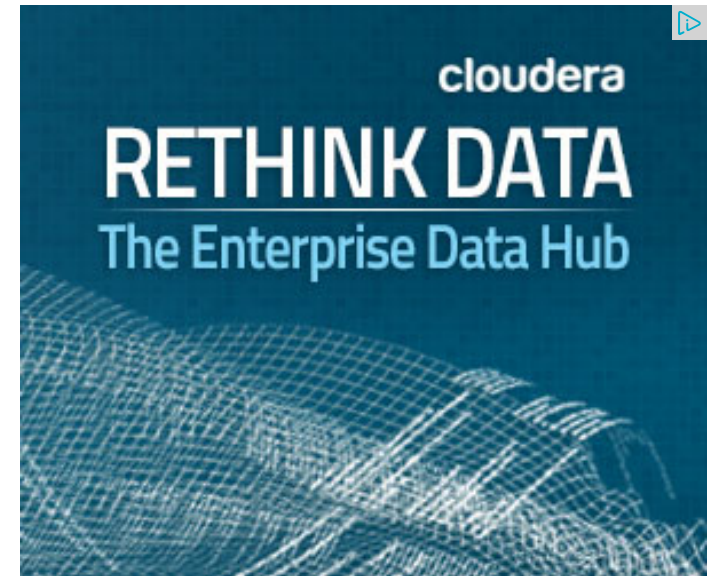
Pattern Searching

Divide & Conquer

Mathematical Algorithms

Recursion

Geometric Algorithms



Popular Posts

All permutations of a given string

Memory Layout of C Programs

Understanding "extern" keyword in C

Median of two sorted arrays

Tree traversal without recursion and without stack!

Structure Member Alignment, Padding and

Data Packing

Intersection point of two Linked Lists

Lowest Common Ancestor in a BST.

Check if a binary tree is BST or not

Sorted Linked List to Balanced BST

```
/* Function to swap values at two pointers */
```

```
void swap (char *x, char *y)
```

```
{
    char temp;
    temp = *x;
    *x = *y;
    *y = temp;
}
```

```
/* Function to print permutations of string
```

```
This function takes three parameters:
```

```
1. String
```

```
2. Starting index of the string
```

```
3. Ending index of the string. */
```

```
void permute(char *a, int i, int n)
```

```
{
    int j;
    if (i == n)
        printf("%s\n", a);
    else
    {
        for (j = i; j <= n; j++)
        {
            swap((a+i), (a+j));
            permute(a, i+1, n);
            swap((a+i), (a+j)); //backtrack
        }
    }
}
```

```
/* Driver program to test above functions */
```

```
int main()
```

```
{
    char a[] = "ABC";
    permute(a, 0, 2);
    getchar();
    return 0;
}
```

Output:

ABC

ACB

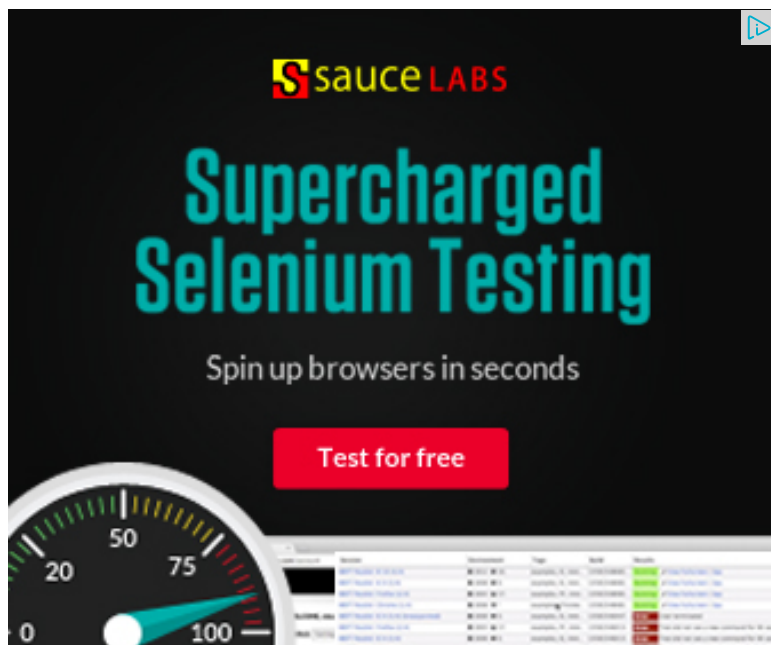
BAC

BCA

Algorithm Paradigm: Backtracking

Time Complexity: $O(n \cdot n!)$

Please write comments if you find the above codes/algorithms incorrect, or find other ways to solve the same problem.



Related Topics:

- Print all possible words from phone digits
- Printing Longest Common Subsequence
- Suffix Array | Set 2 (nLogn Algorithm)
- Rearrange a string so that all same characters become d distance away
- Recursively remove all adjacent duplicates
- Find the first non-repeating character from a stream of characters
- Dynamic Programming | Set 33 (Find if a string is interleaved of two other strings)



- Remove "b" and "ac" from a given string



113



Tweet

1



10



705



Subscribe

Writing code in comment? Please use ideone.com and share the link here.

Recent Comments

Abhi You live US or India?

[Google \(Mountain View\) interview](#) · 8 minutes ago

Aman Hi, Why arent we checking for conditions...

[Write a C program to Delete a Tree.](#) · 47 minutes ago

kzs please provide solution for the problem...

[Backtracking | Set 2 \(Rat in a Maze\)](#) · 51 minutes ago

Sanjay Agarwal bool

tree::Root_to_leaf_path_given_sum(tree...

[Root to leaf path sum equal to a given number](#) · 1 hour ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...

[Count trailing zeroes in factorial of a number](#) · 1 hour ago

newCoder3006 If the array contains negative numbers also. We...


[Find subarray with given sum](#) · 1 hour ago

AdChoices

[► String Function](#)

[► C++ at String](#)


[► String Java](#)

AdChoices 

► [C String](#)

► [String String](#)

► [String 0](#)

AdChoices 

► [String Search](#)

► [For String](#)

► [The String](#)

@geeksforgeeks, **Some rights reserved**

[Contact Us!](#)

Powered by **WordPress** & **MooTools**, customized by geeksforgeeks team