

Subarray/Substring vs Subsequence and Programs to Generate them

Subarray/Substring

A subarray is a **contiguous** part of array. An array that is inside another array. For example, consider the array [1, 2, 3, 4], There are 10 non-empty sub-arrays. The subarrays are (1), (2), (3), (4), (1,2), (2,3), (3,4), (1,2,3), (2,3,4) and (1,2,3,4). In general, for an array/string of size n, there are $n*(n+1)/2$ non-empty subarrays/substrings.

How to generate all subarrays?

We can run two nested loops, the outer loop picks starting element and inner loop considers all elements on right of the picked elements as ending element of subarray.

```
/* C++ code to generate all possible subarrays/subArrays
   Complexity-  $O(n^3)$  */
#include<bits/stdc++.h>
using namespace std;

// Prints all subarrays in arr[0..n-1]
void subArray(int arr[], int n)
{
    // Pick starting point
    for (int i=0; i < n; i++)
    {
        // Pick ending point
        for (int j=i; j<n; j++)
        {
            // Print subarray between current starting
            // and ending points
            for (int k=i; k<=j; k++)
                cout << arr[k] << " ";

            cout << endl;
        }
    }
}

// Driver program
int main()
{
    int arr[] = {1, 2, 3, 4};
    int n = sizeof(arr)/sizeof(arr[0]);
    cout << "All Non-empty Subarrays\n";
    subArray(arr, n);
    return 0;
}
```

Output:

```
All Non-empty Subarrays
1
1 2
1 2 3
1 2 3 4
2
2 3
2 3 4
3
3 4
4
```

Subsequence

A subsequence is a sequence that can be derived from another sequence by zero or more elements, without changing the order of the remaining elements.

For the same example, there are 15 sub-sequences. They are (1), (2), (3), (4), (1,2), (1,3),(1,4), (2,3), (2,4), (3,4), (1,2,3), (1,2,4), (1,3,4), (2,3,4), (1,2,3,4). More generally, we can say that for a sequence of size n , we can have $(2^n - 1)$ non-empty sub-sequences in total.

A string example to differentiate: Consider strings “geeksforgeeks” and “gks”. “gks” is a subsequence of “geeksforgeeks” but not a substring. “geeks” is both a subsequence and subarray. Every subarray is a subsequence. More specifically, **Subsequence is a generalization of substring.**

How to generate all Subsequences?

We can use [algorithm to generate power set](#) for generation of all subsequences.

```
/* C++ code to generate all possible subsequences.
   Time Complexity O(n * 2^n) */
#include<bits/stdc++.h>
using namespace std;

void printSubsequences(int arr[], int n)
{
    /* Number of subsequences is (2**n -1)*/
    unsigned int opsize = pow(2, n);

    /* Run from counter 000..1 to 111..1*/
    for (int counter = 1; counter < opsize; counter++)
    {
        for (int j = 0; j < n; j++)
        {
            /* Check if jth bit in the counter is set
               If set then print jth element from arr[] */
            if (counter & (1<<j))
                cout << arr[j] << " ";
        }
        cout << endl;
    }
}

// Driver program
int main()
{
    int arr[] = {1, 2, 3, 4};
    int n = sizeof(arr)/sizeof(arr[0]);
    cout << "All Non-empty Subsequences\n";
    printSubsequences(arr, n);
    return 0;
}
```

All Non-empty Subsequences

```
1
2
1 2
3
1 3
2 3
1 2 3
4
1 4
2 4
1 2 4
3 4
1 3 4
2 3 4
1 2 3 4
```