Difference between Subarray, Subsequence, and Subset

This post will discuss the difference between a subarray, a substring, a subsequence, and a subset.

1. Subarray

A subarray is a slice from a contiguous array (i.e., occupy consecutive positions) and inherently maintains the order of elements. For example, the subarrays of array `{1, 2, 3}` are `{1}`, `{1, 2}`, `{1, 2, 3}`, `{2}`, `{2, 3}`, and `{3}`.

Following is the C, Java, and Python program to generate all subarrays of the specified array:

C

```
1  #include <stdio.h>
2
3  // Function to print a subarray formed by `[start, end]`
4  void printSubarray(int arr[], int start, int end)
5  {
6      printf("[");
7
8      for (int i = start; i < end; i++) {
9          printf("%d, ", arr[i]);
```

```c
    printf("%d]\n", arr[end]);
}

// Function to print all subarrays of the specified array
void printAllSubarrays(int arr[], int n)
{
    // consider all subarrays starting from `i`
    for (int i = 0; i < n; i++)
    {
        // consider all subarrays ending at `j`
        for (int j = i; j < n; j++) {
            printSubarray(arr, i, j);
        }
    }
}

int main()
{
    int arr[] = { 1, 2, 3, 4, 5 };
    int n = sizeof(arr)/sizeof(arr[0]);

    printAllSubarrays(arr, n);

    return 0;
}
```

Download Run Code

Java

Python

Output:

```
[1]
[1, 2]
[1, 2, 3]
[1, 2, 3, 4]
[1, 2, 3, 4, 5]
[2]
[2, 3]
[2, 3, 4]
[2, 3, 4, 5]
[3]
```

```
[3, 4, 5]
[4]
[4, 5]
[5]
```

Please note that there are precisely `n×(n+1)/2` subarrays in an array of size `n`. Also, there is no such thing as a contiguous subarray. The prefix contiguous is sometimes applied to make the context more clear. So, a contiguous subarray is just another name for a subarray.

2. Substring

A substring of a string `s` is a string `s'` that occurs in `s`. A substring is almost similar to a subarray, but it is in the context of strings.

For example, the substrings of string `'apple'` are `'apple', 'appl', 'pple', 'app', 'ppl', 'ple', 'ap', 'pp', 'pl', 'le', 'a', 'p', 'l', 'e', ''`. Following is the C++, Java, and Python program that generates all non-empty substrings of the specified string:

C++

```
#include <iostream>
using namespace std;

// Function to print all non-empty substrings of the specified string
void printAllSubstrings(string str)
{
    int n = str.length();

    // consider all substrings starting from `i`
    for (int i = 0; i < n; i++)
    {
        // consider all substrings ending at `j`
        for (int j = i; j < n; j++) {
            cout << "'" << str.substr(i, j - i + 1) << "', ";
        }
    }
```

```
19  int main()
20  {
21      string str = "techie";
22      printAllSubstrings(str);
23
24      return 0;
25  }
```

Download Run Code

Output:

```
't', 'te', 'tec', 'tech', 'techi', 'techie', 'e', 'ec', 'ech',
'echi', 'echie', 'c', 'ch', 'chi', 'chie', 'h', 'hi', 'hie', 'i',
'ie', 'e'
```

Java

Python

3. Subsequence

A subsequence is a sequence that can be derived from another sequence by deleting some elements without changing the order of the remaining elements. For example, `{A, B, D}` is a subsequence of sequence `{A, B, C, D, E}` obtained after removing `{C}` and `{E}`.

People are often confused between a subarray/substring and a subsequence. A subarray or substring will always be contiguous, but a subsequence need not be contiguous. That is, subsequences are not required to occupy consecutive positions within the original sequences. But we can say that both contiguous subsequence and subarray are the same.

In other words, the subsequence is a generalization of a substring, or substring is a refinement of the subsequence. For example, `{A, C, E}` is a subsequence of `{A, B, C, D, E}`, but not a substring, and `{A, B, C}` is both a subarray and a subsequence.

given set, `S`, we can find the power set by generating all binary numbers between `0` and 2^n-1, where `n` is the size of the given set. This approach is demonstrated below in C++, Java, and Python:

C++

```
#include <iostream>
#include <cmath>
using namespace std;

// Function to print all subsequences of the specified string
void findPowerSet(string str)
{
    int n = str.length();

    // `N` stores the total number of subsets
    int N = pow(2, n);

    // generate each subset one by one
    for (int i = 0; i < N; i++)
    {
        cout << "'";

        // check every bit of `i`
        for (int j = 0; j < n; j++)
        {
            // if j'th bit of `i` is set, print `S[j]`
            if (i & (1 << j)) {
                cout << str[j];
            }
        }
        cout << "', ";
    }
}

int main()
{
    string str = "apple";
    findPowerSet(str);

    return 0;
}
```

Download Run Code

Output:

```
'pl', 'apl', 'ppl', 'appl', 'e', 'ae', 'pe', 'ape', 'pe', 'ape',
'ppe', 'appe', 'le', 'ale', 'ple', 'aple', 'ple', 'aple', 'pple',
'apple'
```

Java

Python

4. Subset

A subset is any possible combination of the original set. The term subset is often used for subsequence, but that's not right. A subsequence always maintains the relative order of the array elements (i.e., increasing index), but there is no such restriction on a subset. For example, `{3, 1}` is a valid subset of `{1, 2, 3, 4, 5}`, but it is neither a subsequence nor a subarray.

It is worth noting that all subarrays are subsequences and all subsequences are a subset, but the reverse is not valid. For instance, a subarray `{1, 2}` of array `{1, 2, 3, 4, 5}` is also a subsequence and a subset.

Array, Basic, String

Beginner, Must Know