

# Strings

In this lesson, we'll discuss common string terms before moving onto problems.

## We'll cover the following ^

- String
- Anagram
- Substring
- Subsequence
- Palindrome

## String #

A string is a sequence of characters; hence it can be represented using an array of characters.

C++ also has a `string` type that can be used instead of `char[]`.

First, let's go over some common terms used in programming competitions for strings.

---

## Anagram #

The anagram of a string is obtained by rearranging the letters in the string.

For example: `evil` is an anagram of `live`, other anagrams are `leiv` and `vile`.

Whereas `lie` is not because we removed `v`.

---

## Substring #

Substring is a **contiguous** sequence of characters within a string. Analogous to **subarray** in arrays.

For example: for the string `competitive`

For example, for the string `competitive`

`comp`, `tive`, and `pet` are substrings. `cope` is not because it's not contiguous in `competitive`.

Correspondingly, there are  $\left[\frac{N \times (N-1)}{2} + N\right]$  substrings in a string of length  $N$ .

---

## Subsequence #

A subsequence of a sequence is obtained by deleting some or none of the elements without changing the order of elements. Similar to what we discussed in arrays.

For example, for the string `competitive`:

- `cope` and `pit` are subsequences but `tom` is not.

There are  $2^N$  subsequences of a string of length  $N$ .

---

## Palindrome #

Palindromes are simply strings that read the same forwards and backwards.

For example: `madam` or `racecar` is the same spelt backwards.

---

In the next lesson, we'll see how to use C++ STL string.