1859. Sorting the Sentence

A sentence is a list of words that are separated by a single space with no leading or trailing spaces. Each word consists of lowercase and uppercase English letters.

A sentence can be shuffled by appending the 1-indexed word position to each word then rearranging the words in the sentence.

For example, the sentence "This is a sentence" can be shuffled as "sentence4 a3 is2 This1" or "is2 sentence4 This1 a3".
Given a shuffled sentence s containing no more than 9 words, reconstruct and return the original sentence.

Example 1:

Input: s = "is2 sentence4 This1 a3"
Output: "This is a sentence"
Explanation: Sort the words in s to their original positions "This1 is2 a3 sentence4", then remove the numbers.

Example 2:

Input: s = "Myself2 Me1 I4 and3"
Output: "Me Myself and I"
Explanation: Sort the words in s to their original positions "Me1 Myself2 and3 I4", then remove the numbers.

Constraints:

2 <= s.length <= 200
s consists of lowercase and uppercase English letters, spaces, and digits from 1 to 9.
The number of words in s is between 1 and 9.
The words in s are separated by a single space.
s contains no leading or trailing spaces.

Code

C++ Auto

```cpp
class Solution {
public:
    string sortSentence(string s) {

    string str,my_s;
    int words = 1;
    int s_length = s.length();


    stringstream ss(s);

    for (size_t i = 0; i < s_length; i++)
    {
        if(s[i] == ' '){
            words++;
        }
    }

    string my_array[words];

    while (getline(ss, str, ' ')) {
        my_array[int(str.back()) - 49] = str.substr(0, str.size()-1);
    }

    for (auto value : my_array)
    {
        my_s.append(value).append(" ");
    }

    return my_s.substr(0, my_s.size()-1);

    }

};
```

Sources:

https://www.geeksforgeeks.org/cpp-program-for-char-to-int-conversion/

If a numeric character needs to be typecasted into the integer value then either we can subtract 48 or '0' and then typecast the numeric character into int.

Below is the C++ program to convert char to integer value using typecasting:

C++

```cpp
// C++ program to convert
// char to int (integer value) using typecasting
#include <iostream>
using namespace std;

// Driver code
int main()
{
    char ch = '5';

    // Subtracting 48 will produce desired results
    cout << int(ch) - 48 << "\n";

    // Also subtracting '0' will result in same output
    cout << int(ch - '0');
    return 0;
}

// This code is contributed by Susobhan Akhuli
```

https://www.digitalocean.com/community/tutorials/string-concatenation-in-c-plus-plus

3. The append() Method for String Concatenation in C++

C++ has another built-in method: **append()** to concatenate strings. The `append()` method can be used to add strings together. It takes a string as a parameter and adds it to the end of the other string object.

Syntax:

```
string1.append(string2);
```

Example:

```
#include <bits/stdc++.h>
using namespace std;

int main()
{   string str1="", str2="";

    cout<<"Enter String 1:\n";
    cin>>str1;
    cout<<"Enter String 2:\n";
    cin>>str2;

    str1.append(str2);
    cout<<"Concatenated String:"<<endl;
    cout<<str1;
    return 0;
}
```

https://stackoverflow.com/questions/2310939/remove-last-character-from-c-string

262

For a non-mutating version:

```
st = myString.substr(0, myString.size()-1);
```

Share Improve this answer Follow

answered Feb 22, 2010 at 12:59

Matthieu M.

292k • 49 • 462 • 737

https://favtutor.com/blogs/split-string-cpp

5) Using std::getline() Function

Another method to split strings in C++ is by using the std:::getline() function. This function reads a string from an input stream until a delimiter character is encountered. Just as we take the input from the user using getline() function, similarly we will take the input into the sringstream using getline() function.

Syntax:

```
getline(string, token, delimiter);
```

Here, the token saves the extracted string tokens from the original string. Below is the C++ program implementation:

```
// Welcome to favtutor
#include <bits/stdc++.h>
using namespace std;

int main() {
    string s, str;

    s = "I love to read articles on Favtutor.";

    // ss is an object of stringstream that references the S string.
    stringstream ss(s);

    // Use while loop to check the getline() function condition.
    while (getline(ss, str, ' '))
        // `str` is used to store the token string while ' ' whitespace is used as the del
        cout << str << endl;

    return 0;
}
```

Output:

```
I
love
to
read
articles
on
Favtutor.
```