Лабораторная №11. Строки класса string. Алгоритмы обработки текста.

Темы, рассматриваемые в лабораторной работе:

Использование методов класса std::string. Базовые алгоритмы обработки текста (поиск, вставка, удаление части строки, удаление незначащих символов, выделение лексем и т.д.)

При решении следующих задач демонстрируется использование строк `std::string`

```
#include <iostream>       // std::cout
#include <string>         // std::string
// использование
// std::string::find()
// std::string::npos
// std::string::replace()
// std::string::rfind()


int main ()
{
  std::string str ("There are two needles in this haystack with needles.");
  std::string str2 ("needle");

  // different member versions of find in the same order as above:
  std::size_t found = str.find(str2);
  if (found!=std::string::npos)
    std::cout << "first 'needle' found at: " << found << '\n';

  found=str.find("needles are small",found+1,6);
  if (found!=std::string::npos)
    std::cout << "second 'needle' found at: " << found << '\n';

  found=str.find("haystack");
  if (found!=std::string::npos)
    std::cout << "'haystack' also found at: " << found << '\n';

  found=str.find('.');
  if (found!=std::string::npos)
    std::cout << "Period found at: " << found << '\n';

  // let's replace the first needle:
  str.replace(str.find(str2),str2.length(),"preposition");
  std::cout << str << '\n';

  std::string str3 ("The sixth sick sheik's sixth sheep's sick.");
  std::string key ("sixth");

  found = str3.rfind(key);
  if (found!=std::string::npos)
    str3.replace (found,key.length(),"seventh");

  std::cout << str3 << '\n';

  return 0;
}

#include <iostream>
#include <string>
// использование
```

```
// std::string::substr()

int main ()
{
  std::string str="We think in generalities, but we live in details.";
 // (quoting Alfred N. Whitehead)
  std::string str2 = str.substr (3,5);     // "think"
  std::size_t pos = str.find("live");      // position of "live" in str
  std::string str3 = str.substr (pos);     // get from "live" to the end
  std::cout << str2 << ' ' << str3 << '\n';

  return 0;
}
```

Задания:

1. Дана строка, состоящая из слов, разделенных одним или несколькими пробелами. Посчитать количество слов в строке.
2. Удалить из строки все вхождения заданного символа.
3. Вставить в строку другую строку после первого вхождения заданного символа.
4. Дана строка, состоящая из слов, разделенных одним или несколькими пробелами. Найти самое длинное слово (первое, если их несколько) и удалить его из строки.
5. Дана строка, состоящая из слов, разделенных одним или несколькими пробелами. Заменить в строке каждое четное слово заданным словом.

Дополнительные задания:

6. Дана строка, состоящая из слов, разделенных одним или несколькими пробелами. Каждое нечетное слово записать в строке в обратном порядке (перевернуть слово).
7. Дана строка, состоящая из слов, разделенных одним или несколькими пробелами и знаками препинания. Выполнить корректировку расстановки пробелов – между словами должно быть по одному пробелу (лишние убрать), после знака препинания должен быть 1 пробел (добавить отсутствующие или удалить лишние).