# Задание по программированию: Группировка строк по префиксу

# Условие

# Часть 1. Группировка по символу

Напишите функцию FindStartsWith:

* принимающую отсортированный набор строк в виде итераторов range\_begin, range\_end и один символ prefix;
* возвращающую диапазон строк, начинающихся с символа prefix, в виде пары итераторов.

template <typename RandomIt>

pair<RandomIt, RandomIt> FindStartsWith(

    RandomIt range\_begin, RandomIt range\_end,

    char prefix);

Если итоговый диапазон пуст, его границы должны указывать на то место в контейнере, куда можно без нарушения порядка сортировки вставить любую строку, начинающуюся с символа prefix (подобно алгоритму equal\_range). Гарантируется, что строки состоят лишь из строчных латинских букв и символ prefix также является строчной латинской буквой.

Поиск должен осуществляться за логарифмическую сложность — например, с помощью двоичного поиска.

## Пример кода

int main() {

  const vector<string> sorted\_strings = {"moscow", "murmansk", "vologda"};

  const auto m\_result =

      FindStartsWith(begin(sorted\_strings), end(sorted\_strings), 'm');

  for (auto it = m\_result.first; it != m\_result.second; ++it) {

    cout << \*it << " ";

  }

  cout << endl;

  const auto p\_result =

      FindStartsWith(begin(sorted\_strings), end(sorted\_strings), 'p');

  cout << (p\_result.first - begin(sorted\_strings)) << " " <<

      (p\_result.second - begin(sorted\_strings)) << endl;

  const auto z\_result =

      FindStartsWith(begin(sorted\_strings), end(sorted\_strings), 'z');

  cout << (z\_result.first - begin(sorted\_strings)) << " " <<

      (z\_result.second - begin(sorted\_strings)) << endl;

  return 0;

}

### Вывод

moscow murmansk

2 2

3 3

## Подсказка

К символам (char) можно прибавлять или вычитать числа, получая таким образом следующие или предыдущие буквы в алфавитном порядке. Например, для строки s выражение --s[0] заменит её первую букву на предыдущую.

Обратите внимание, что выражение 'a' + 1 имеет тип int и поэтому может понадобиться привести его к типу char с помощью static\_cast.

# Часть 2. Группировка по префиксу

Напишите более универсальный вариант функции FindStartsWith, принимающий в качестве префикса произвольную строку, состоящую из строчных латинских букв.

template <typename RandomIt>

pair<RandomIt, RandomIt> FindStartsWith(

    RandomIt range\_begin, RandomIt range\_end,

    const string& prefix);

## Пример кода

int main() {

  const vector<string> sorted\_strings = {"moscow", "motovilikha", "murmansk"};

  const auto mo\_result =

      FindStartsWith(begin(sorted\_strings), end(sorted\_strings), "mo");

  for (auto it = mo\_result.first; it != mo\_result.second; ++it) {

    cout << \*it << " ";

  }

  cout << endl;

  const auto mt\_result =

      FindStartsWith(begin(sorted\_strings), end(sorted\_strings), "mt");

  cout << (mt\_result.first - begin(sorted\_strings)) << " " <<

      (mt\_result.second - begin(sorted\_strings)) << endl;

  const auto na\_result =

      FindStartsWith(begin(sorted\_strings), end(sorted\_strings), "na");

  cout << (na\_result.first - begin(sorted\_strings)) << " " <<

      (na\_result.second - begin(sorted\_strings)) << endl;

  return 0;

}

### Вывод

moscow motovilikha

2 2

3 3