

## ОПИСАНИЕ ЗАДАНИЯ

В этой лабораторной работе от вас требуется продемонстрировать умение разрабатывать код с использованием контейнеров библиотеки STL и оценить производительность кода вывода данных стандартной библиотеки.

### ЗАДАНИЕ ДЛЯ ВЫБРАВШИХ БАЗОВЫЙ УРОВЕНЬ

1. показать умение работать с векторами: объявление, добавление элементов в вектор, вставка и удаление элементов из вектора; применение алгоритмов `for_each` и `random_shuffle` при работе с векторами;
2. привести результат сравнения времени вывода 10 000 элементов типа `char`, `int`, `double` и строк `char []` с помощью функций `printf` (ANSI C) и `std::cout` (C++), результат представить в виде таблицы;

	char	int	double	"PKIMS RULEZ"
<code>printf</code>				
<code>std::cout</code>				

3. разработать код функции сортировки вставками (вариант 2 из лекций) и для целочисленных векторов размером 3, 5 и 10 элементов привести результаты сравнения быстродействия разработанной функции и реализованного в стандартной библиотеке алгоритма `std::sort`.

	3 элемента	5 элементов	10 элементов
<code>insertion_sort_2</code>			
<code>std::sort</code>			

#### За что снимаются баллы на базовом уровне

1. код не отформатирован, нагромождение конструкций: -1 балл
2. время выдаётся в неадекватном виде, которое ничего не говорит о быстродействии кода: -1 балл
3. вы не можете пояснить, как работает код с первого раза: -2 балла (но вы всё равно должны будете в итоге объяснить, как он работает, иначе лаба не будет зачтена)

### ЗАДАНИЕ ДЛЯ ВЫБРАВШИХ ПОВЫШЕННЫЙ УРОВЕНЬ

Необходимо выполнить все задания на минимум. После этого необходимо выполнить дополнительные задания, приведённые ниже.

*Эти задания выполняются последовательно.*

- **Плюс балл.** Самостоятельно реализовать аналог класса `vector` для динамического массива целых чисел, который имеет как минимум следующие функции-члены класса:

`push_back(int val)` – для добавления элемента `val`;

`pop_back()` – для извлечения (удаления) последнего элемента;

back() – для доступа к последнему элементу;  
size() – для получения размера вектора;  
insert(int pos, int val) – для вставки элемента val на позицию pos;  
clear() – для очистки вектора;  
erase(int pos) – для удаления элемента на позиции pos;  
erase(int from, int to) – для удаления элемента с позиции from до позиции to;

- **Плюс балл.** Модифицировать предыдущий вариант так, чтобы вместо целых чисел использовался шаблон (template).

#### **За что снимаются баллы на повышенном уровне**

1. код не отформатирован, нагромождение конструкций: -1 балл;
2. время выдаётся в неадекватном виде, которое ничего не говорит о быстродействии кода: -1 балл;
3. не показано умение пользоваться спецификаторами доступа в классе: -1 балл;
4. вы не можете пояснить, как работает код с первого раза: -4 балла (но вы всё равно должны будете в итоге объяснить, как он работает, иначе лаба не будет зачтена)