

Знакомство

Знакомство

- **Завадский Евгений Владимирович**
- Связаться: tg **@ev0913**
- Тg-группа с актуальной информацией

План занятий

Курс состоит из 3 лабораторных работ:

- Класса String (STL-like)
- UML-диаграмма + рефакторинг кода (C -> C++)
- Текстовый редактор (MVC)

Для сдачи ЛР необходимо:

- 1) выполнить задание ЛР
- 2) написать отчет о проделанной работе
- 3) написать тест

На выполнение одной работы дается 4-5 недель

Тестирование

- Необходимо для сдачи работы
- Состоит из вопросов по лабораторной работе и лекционному материалу
- В случае плохого результата выполняется работа над ошибками – студент выполняет детальный разбор вопросов с неправильными ответами и объясняет их преподавателю
- Пишется во время занятий за неделю до дедлайна

Отчет

- Необходим для сдачи работы
- Должен включать:
 - Обязательные пункты, которые описаны в задании к ЛР в разделе Report's content
- Должен быть написан на английском языке

Лабораторная работа

- Лабораторные сдаются в свободное время на занятии
- За один подход можно показать только одну лабораторную (RoundRobin)
- По каждой работе установлен дедлайн

Дополнительные задания

- У каждой работы есть список дополнительных заданий
- Выполненное дополнительное задание = «+»
- В конце семестра «+» конвертируются в дополнительные баллы к итоговому тесту

Расписание

Виды занятий:

- теоретический материал по ЛР
- практическая работа
- дополнительный материал по курсу
- тестирование
- прием лабораторных работ

Первая лабораторная

Класс String

Класс и объекты

Ссылки и указатели

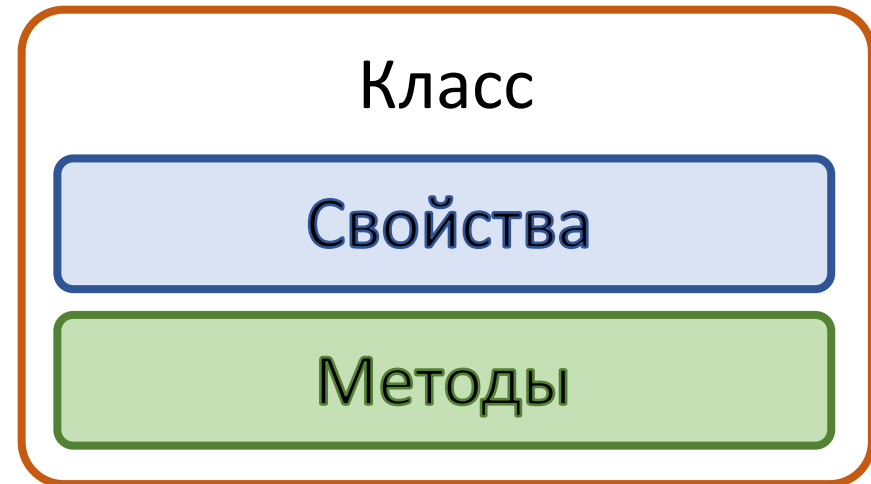
Move семантика

Итераторы

Использование C++ класса в Python

Класс и объект

```
1  class Location {
2      Location(std::string name): name_(name) {};
3      std::string name_;
4  }
5
6  class Mammal {
7  protected:
8      Mammal() {}
9      Mammal(std::string, Location);
10 public:
11     void move_around(int, int);
12     void migrate(Location);
13     virtual void make_sound() = 0;
14     virtual void print_info();
15
16 protected:
17     Location location_;
18 private:
19     std::string name_;
20 };
21
22 class Dog : public Mammal {
23 public:
24     Dog(std::string name, Location location): Mammal(name, location) {}
25     virtual void make_sound() override;
26 };
```



Объект – экземпляр класса

```
Dog spyke("Spyke", Location("Home"));
```

STL-контейнер

- **STL** – стандартная библиотека шаблонов
- **Контейнер** – шаблон класса, обеспечивающего хранение коллекций связанных объектов

```
typedef basic_string<char> string  
typedef basic_string<wchar_t> string
```

- **Типы контейнеров:**
 - последовательные (list, deque, vector...)
 - ассоциативные (set, map...)
 - адаптеры – разновидность последовательного или ассоциативного с упрощённым интерфейсом (queue, stack)

Основные этапы работы

Подготовительная часть:

- 1) Изучить требуемый стиль написания кода (приведен в файле CODE_STYLE.docx)
- 2) Проанализировать конструкторы, операторы и методы. Цель – выделить их общее поведение и в основной части не писать 100500 одинаковых конструкторов, операторов и методов, а переиспользовать уже написанный код.

В рамках **основной части** необходимо:

- 3) Объявить класс MyString в заголовочном файле mystring.h
- 4) Определить конструкторы, операторы и методы в файле mystring.cpp

При разработке класса **стоит учитывать:**

- 1) Использовать **std::string** под капотом MyString **запрещено**
- 2) Использовать **Си-функции**, которые заменены в C++, **запрещено** (printf ->std::cout)
- 3) Размер «буфера» изменяется в меньшую сторону только при помощи метода shrink_to_fit
- 4) Метод shrink_to_fit делает размер «буфера» равным длине хранимой строки
- 5) Метод shrink_to_fit предназначен только для пользователя и не должен вызываться в конструкторах, операторах и методах
- 6) При возникновении вопросов о возвращаемых методами и операторами значениях можно изучить их объявления для std::string (<https://cplusplus.com/reference/string/string/>)

Календарь

- Первая лабораторная:
 - Тест – **27 сентября**
 - Дедлайн – **4 октября**

- Вторая лабораторная:
 - Тест – **25 октября**
 - Дедлайн – **1 ноября**

- Третья лабораторная:
 - Тест – **29 ноября**
 - Дедлайн – **6 декабря**