# Аггрегация, компазиция, наследование

# Материалы:

<https://ru.wikipedia.org/wiki/Агрегирование_(программирование)>

<https://ru.stackoverflow.com/questions/454900/Агрегация-между-классами>

<https://ru.wikipedia.org/wiki/Наследование_(программирование)>

<https://msdn.microsoft.com/ru-ru/library/a48h1tew.aspx>

<https://habrahabr.ru/post/177447/>

<http://cppstudio.com/post/10103/>

## Дополнительно про виртуальность:

<http://en.cppreference.com/w/cpp/language/abstract_class>

<https://msdn.microsoft.com/en-us/library/0y01k918.aspx>

<https://habrahabr.ru/post/51229/>

# Задания в классе:

1. Создайте класс Student, который будет содержать информацию о студенте. С помощью механизма наследования реализуйте класс Aspirant (аспирант – студент, который готовится к защите кандидатской работы) производный от Student.
2. Создать абстрактный базовый класс «Транспортное средство» и производные классы «Автомобиль», «Велосипед», «Повозка».  
   Подсчитать время и стоимость перевозки пассажиров и грузов каждым транспортным средством.

# Домашнее задание:

Описать базовый класс «Строка».

Методы:

* конструктор без параметров;
* конструктор, принимающий в качестве параметра C-строку (заканчивается нулевым байтом);
* конструктор копирования;
* оператор присваивания;
* получение длины строки;
* очистка строки (сделать строку пустой);
* деструктор;
* конкатенация строк (перегрузить операторы + и +=);
* проверка на равенство (= =) и на неравенство (!=).

Описать класс «Битовая строка» производный от класса «Строка» (Строки данного класса могут содержать только символы ‘0’ и ‘1’). Если в основе инициализирующей строки встретятся любые символы, отличные от допустимых, то «Битовая строка» становится пустой. Содержимое строки рассматривается как двоичное представление целого числа со знаковым разрядом.

Методы, которые переопределены:

* конструктор, принимающий в качестве параметра C-строку (он должен проверять строку на наличие лишних символов);
* конструктор копирования (если к нам поступает класс String, наш базовый, то мы также должны проверить строку на лишние символы);
* оператор присваивания (тоже самое, что и конструктор копирования);
* сложение битовых строк (перегрузить операторы + и +=);
* проверка на равенство (= =) и на неравенство (!=).

Дополнительно:

Можно реализовать работу со строкой как с числом, то есть операции +, -, +=, -= теперь будут работать математически. Также можно обрабатывать отрицательные числа, используя дополнительный код или маркер.

<https://ru.wikipedia.org/wiki/Дополнительный_код>