

## 1. Строка

*Напишите класс строки.*

*Предусмотрите возможность ввода-вывода из потока, обращения по индексу, конструктора из `char*`, `std::string`, операторов присваивания, сравнения, сложения, обращения к внутреннему содержимому, вставки в середину строки, удаления из середины, вставки в конец, удаления последней буквы, умному выделению и освобождению памяти.*

Возможность обращения к внутреннему содержимому — это метод, возвращающий указатель на с-строку:

```
const char *c_str() const;
```

Возможность вставки в середину строки должна обеспечиваться методом следующего прототипа:

```
void insert(int index, char ch);
```

где `index` — индекс буквы, перед которой предполагается вставить символ `ch`.

Удаление из середины должно обеспечиваться методом следующего прототипа:

```
void delete(int firstChar, int length);
```

— удаляет `length` символов, начиная с `firstChar`.

Вставка в конец:

```
void append(char ch);
```

— добавляет символ `ch` в конец строки.

Удаление последней буквы:

```
void deleteLast();
```

Необходимо по возможности минимизировать число обращений к `new` и `delete`. Для этого придется хранить в классе объем текущей выделенной памяти, который не обязан соответствовать длине хранимой строки (но, естественно, должен быть не менее этой длины).

Также нужно хранить в классе собственно длину строки (теоретически без этого можно обойтись, но тогда многие операции сильно замедлятся).

В функции `main` проверить работоспособность всех конструкторов, методов и операций.

## 2. Обращение

*Написать чисто виртуальный класс `Human`, который содержит поля `firstName` и `secondName` и виртуальный метод `getTitle() const`. Отнаследовать от него классы `Mister`, `Miss`, `KId`, в которых будет переопределена реализация `getTitle` на возвращение строк `"Mr. secondName"`, `"Ms. secondName"`, `"firstName"` соответственно.*

Использовать класс Строка из задачи 1. Не использовать стандартный класс `std::string`.

## 3. Пропускная система

*Есть 6 классов: Здание, Лаборатория, УчебныйКорпус, Человек, Студент, Сотрудник. В классе Здание реализовать метод `bool permission(const Human*) const`, который пропускает в учебное здание всех сотрудников и студентов, в лабораторию пропускает студентов, которые ранее обращались в данную лабораторию за допуском и всех сотрудников. Посторонних людей не пропускают в здания университета вообще.*

Использовать операцию `dynamic_cast`.

Класс Лаборатория должен хранить список допущенных в лабораторию студентов. Также в этом классе должен быть метод, принимающий указатель или ссылку на объект студента, заносающий в список допущенных. Метод `permission()` класса Лаборатория должен, при обращении студента, искать его в списке и, если найдет, пропускать.

Список допущенных можно реализовать на `std::vector` или другом контейнере стандартной библиотеки. Поиск можно не оптимизировать.

#### 4. Библиотека

*Реализуйте класс книги. Книга должна иметь возможность получить и установить информацию о ее авторе или авторах, названии, издательстве, годе выпуска, количестве страниц, содержании.*

*Реализуйте класс для представления информации о собрании сочинений, который хранит в себе объекты "Книга" из предыдущей задачи. Собрание сочинений должно позволять получать/устанавливать информацию об авторе, названии собрания, числе томов, добавлять/удалять книги.*

*Дополните набор классов "Книга" и "Собрание сочинений" необходимыми классами для реализации библиотеки, хранящей отдельные книги, собрания сочинений и периодические (с периодом день, неделя, месяц, год) журналы.*

*Библиотека должна предоставлять методы для добавления, выдачи читателям, получения выданных и удаления литературы из фондов библиотеки, поиска изданий по названиям/авторам/годам выпуска/издательствам. В случае выдачи читателям последнего экземпляра издания про поиске/просмотре экземпляр почемается как имеющийся в библиотеке, но находящийся на руках. При удалении издания - он удаляется и из картотеки.*

*Реализуйте итераторы для просмотра всех изданий, которые зарегистрированы в библиотеке.*

*Имейте в виду, что в библиотеке может быть несколько одинаковых экземпляров изданий.*

Необходимо сделать класс «Издание», и от него производные классы: «Книга», «Собрание сочинений», «Журнал». В классе «Издание» должен быть `public`-метод, возвращающий `bool: true`, если остались экземпляры в наличии и `false`, если все экземпляры на руках.

Класс «Итератор» должен быть подклассом (`embedded class`) класса «Библиотека». При разыменовании итератор должен возвращать класс «Издание».

Использовать класс Строка из задачи 1. Не использовать стандартный класс `std::string`.

#### 5. Геометрические фигуры на плоскости

*Напишите набор классов для представления геометрических фигур на плоскости:*

- Окружность
- Многоугольник
- Выпуклый многоугольник
- Прямоугольник
- Квадрат
- Треугольник

*Эти классы необходимо отнаследовать от базового класса фигуры (*Shape*), в котором определить виртуальные методы для:*

- нахождения площади фигуры
- проверки попадания точки на границу фигуры
- проверки принадлежности точки фигуре
- нахождения точки пересечения фигуры и отрезка

Помимо перечисленных классов, понадобятся также классы «Точка» и «Отрезок».

Все конструкторы должны проверять корректность поданных на вход фигур (нет ли в многоугольнике самопересечений, является ли выпуклый многоугольник действительно выпуклым и т.п.).

Подсказка: классы «Выпуклый многоугольник», «Прямоугольник», «Квадрат», «Треугольник», можно отнаследовать от класса «Многоугольник». Это делать не обязательно, но так сильно уменьшится объем работы.

## 6. Длинные целые

*"Длинное" число необходимо представлять в виде разложения по степеням основания, либо в качестве остатка от деления на простые числа (см. Китайская теорема об остатках).*

*Необходимо реализовать:*

*BigInt();*

*BigInt(int);*

*BigInt(const BigInt&);*

*Арифметические операторы сложения, вычитания, умножения, целочисленного деления и нахождения остатка как на "длинное" число, так и на обычное.*

*Также необходимо реализовать операторы ввода/вывода в поток.*

*Работоспособность продемонстрировать на e-judge:*

<http://acmp.ru/index.asp?main=task&id task=103>

<http://acmp.ru/index.asp?main=task&id task=143>

<http://acmp.ru/index.asp?main=task&id task=144>

<http://acmp.ru/index.asp?main=task&id task=145>

<http://acmp.ru/index.asp?main=task&id task=172>

<http://acmp.ru/index.asp?main=task&id task=363>

Для внутреннего представления использовать такое основание системы счисления, чтобы цифра занимала целое число бит.

## 7. Map Editor

Задача изложена в отдельном файле `map_editor.pdf`