

КУРС C++

ЦАРЬКОВ ОЛЕГ

ОПИСАНИЕ ТИКЕТА OTSN-7

Некоторые стандартные обертки.

В языке `c++` по сравнению с языком `c` добавлено очень много полезных оберток (классов) в пространство имен `std`.

- `std::ofstream` обертка над потоками ввода и вывода. Находится в заголовочном файле `fstream`.

2 конструктора:

- 1) Не принимает никаких переменных, конструируется объект, выводящий на экран с помощью внутреннего оператора `operator <<`.

- 2) Принимает строку. Конструируется объект, выводящий в файл, название которого записано в строке, тоже с помощью оператора `operator <<`.

Пример:

```
int main() {  
    std::ofstream out("file.txt");  
    out << "Text";  
    return 0;  
}
```

Будет создан файл `"text.txt"`, в котором будет написана строка `"Text"`.

- `std::ifstream` обертка над потоками ввода и вывода. Находится в заголовочном файле `fstream`.

Все то же, что и в предыдущем пункте, точно так же можно реализовать чтение данных из файла.

Задание 1.

Написать программу, читающую два числа из файла `"input.txt"` и пишущую их сумму в файл `"output.txt"`.

- `std::pair<typename T1, typename T2>` пара любых объектов.

Конструктор принимает два аргумента — значение первого объекта и значение второго объекта.

Пример

```
int main() {
    std::pair<int, double> pair(3, 1.2);
    return 0;
}
```

Зачем это нужно — трудно сказать.

- `std::vector<typename T>` обертка над массивом. Находится в заголовочном файле *vector*.

Тут имеются:

- 1) Конструктор, не принимающий аргументов. Строится массив размера 0.

- 2) Внутренняя функция *push_back* добавляет элемент в конец.

- 3) Внутренняя функция *pop_back* удаляет элемент из конца и возвращает его.

- 4) Переопределенный оператор *operator[]* принимающий число типа *size_t* и выдающий *i*-ый элемент массива.

- 5) Внутренняя функция *at*, принимающая *i* и возвращающая элемент на *i*-ой позиции

- 6) Внутренняя функция *clear*, очищающая содержимое.

- 7) Внутренняя функция *size*, возвращающая размер.

Пример:

```
int main() {
    std::vector<int> a;
    a.push_back(1);
    a.push_back(2);
    a.push_back(3);
    for(size_t i = 0; i < a.size(); ++i) {
        std::cout << a[i] << " ";
    }
}
```

Программа выведет 123.

Задание 2.

Почему-то не определен *operator <<* для объектов *std::ostream* и *std::vector<typename T>*, то есть нельзя распечатать вектор обычным способом. Определить *operator <<* и *operator >>* для *std::vector*

Задание 3.

Используя *std :: vector*, написать класс многочлена от одной переменной. Многочлены можно складывать, умножать и вычитать. Должна быть введена переменная x , которая равна многочлену первой степени с коэффициентом 1 при x и со всеми остальными коэффициентами, равными нулю.

Должна быть функция печати многочлена на экран. Принятия многочлена с экрана.

Должен быть конструктор, принимающий число и создающий многочлен нулевой степени с таким свободным коэффициентом.