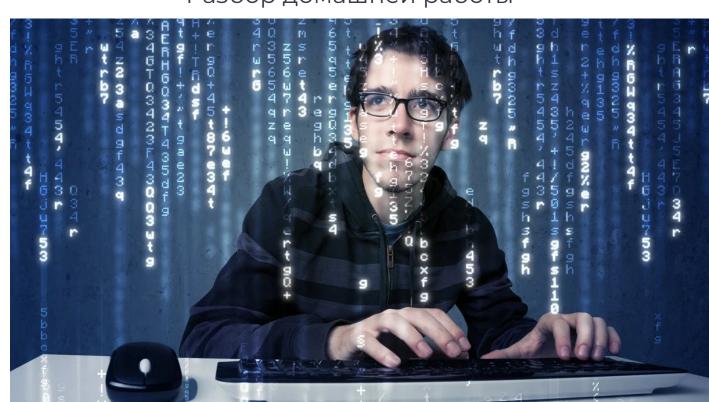


Файлы

Урок №30



Разбор домашней работы



ЧТО HAC ЖДЕТ

ПЛАН УРОКА

- Теория: файловые потоки
- Решаем задачки
- Теория: чтение файла по символам
- Решаем задачки
- Теория: чтение файла по строкам
- Решаем задачки
- Подводим итоги



Теория





Файловые потоки

Заголовок, который необходимо подключить:

#include <fstream>

Два типа данных для файловых потоков:

ifstream - поток ввода

ofstream - поток вывода

Поток ввода

Объявляем переменную для чтения из файла:

```
ifstream input("in.txt");
тип данных название название
переменной файла
```

Теперь мы можем читать из **input** так же как и из **cin**!

```
int a, b;
input >> a >> b;
```

Поток вывода

Объявляем переменную для записи в файл:

```
ofstream output ("out.txt");

тип данных название название
переменной файла
```

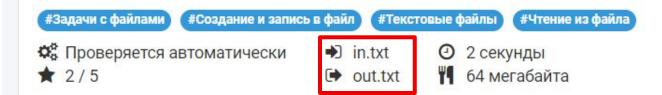
Теперь мы можем писать в **output** так же как и в **cout**!

```
output << "Hello, world!";
output << "a + b = " << (a + b);</pre>
```



Перед практикой





Названия входных и выходных файлов могут быть разными в разных задачах!



Практика







Теория







Чтение символа



Чтение символа

Чтение символа из стандартного потока ввода:

```
char ch = cin.get();

Чтение символа из файла:

ifstream input("in.txt");

char ch = input.get();
```

Чтение символа

Чтение символа из стандартного потока ввода:

```
char ch = cin.get();
```

Чтение символа из файла:

```
fstream input("in.txt");
char ch = input.get();
```

Внимание: в конце файла **input.get()** читает <u>символ, которого нет</u>, поэтому каждый раз после чтения надо проверять, что файл не закончился.



Символ конца файла

Чтобы узнать дошли ли мы до конца файла:

```
bool finished = input.eof();
```

Примечание: eof - end of file

15 ====

Соберём алгоритм чтения файла по символам



```
ifstream input("in.txt");
```

```
ifstream input("in.txt");
char ch = input.get();
```

```
ifstream input("in.txt");
char ch = input.get();
while (!input.eof()) {
    // code
```

```
ifstream input("in.txt");
char ch = input.get();
while (!input.eof()) {
    // code
    ch = input.get();
}
```



Это не очень хорошо, возникает **дублирование** кода!



```
ifstream input("in.txt");
char ch;
while (ch = input.get(), !input.eof()) {
                             Выполняется вторым
          Выполняется первым
                            возвращается результат
         Важно: Последовательность действий
         должна быть именно такой!
```



Практика







Теория





25 ====

Чтение строки целиком

Чтение строки целиком

С помощью какой функции мы считывали всю строку из стандартного потока ввода?

Чтение строки

Чтение строки из стандартного потока ввода:

```
string str;
getline(cin, str);

Чтение строки из файла:
ifstream input("in.txt");
string str;
getline(input, str);
```

Чтение строки

Чтение строки из стандартного потока ввода:

```
string str;
getline(cin, str);

Чтение строки из файла:
ifstream input("in.txt");
string str;
getline(input, str);
```

Внимание: getline (input, str) никогда не читает строки которой нет, поэтому достаточно сначала проверить, что файл не закончился и потом читать строку

Чтение файла по строкам



Практика





- Какой заголовок нужен для работы с потоками файлов?
- 2) В чём разница между ifstream и ofstream?
- 3) Как узнать, дочитали ли мы файл до конца?
- 4) Почему getline() в цикле идёт после eof(), a get() до?