

Code Style

Путевской Александр
гр. ИА-032

February 2022

1 Введение

В данной работе представлен мой стиль написания кода на языках C/C++.

2 Основная часть

Используемые языки программирования

- C/C++;

Названия функций и переменных

Функции и переменные должны иметь описательные и логичные названия, не приветствуются однобуквенные названия (если это не итераторы).

Названия переменных:

```
1 //counter variable:  
2 int m = 0; // wrong name  
3 int counter = 0; // correct name
```

Названия методов и функций должны быть записаны в смешанном регистре и начинаться с нижнего.

Названия методов и функций

```
1 getName();  
2 addFirst();
```

Именованные константы должны быть записаны в верхнем регистре с нижним подчёркиванием в качестве разделителя.

Названия констант

```
1 MAX_ITERATIONS, COLOR_RED
```

Оформление классов

Класс следует объявлять в заголовочном файле и определять (реализовывать) в файле исходного кода, имена файлов совпадают с именем класса:

```
1 MyClass.h, MyClass.cpp
```

Заголовочные файлы объявляют интерфейс, файлы исходного кода его реализовывают.

Не следует объявлять переменные класса как `public`. Вместо этого нужно использовать переменные с модификатором `private` и соответствующие функции доступа:

```
1 class MyClass
2 {
3 public:
4     int getValue () {
5         return value_;
6     }
7     ...
8
9 private:
10    int value_;
11 }
```

Пробелы и отступы

Операторы и операнды разделяются пробелом:

```
1 int x = (a + b) * c / d;
```

Оформление циклов и операторов управления

При использовании циклов и операторов управления всегда используются фигурные скобки, отступы и переносы на новую строку, там где это нужно:

```
1 for (int i=0; i<d; i++){
2     unsigned int q = b[i];
3     for ( ; q!=0; q=q/2){
4         if (q%2==1) c++;
5     }
6 }
```

Цикл `for` используется, когда известно количество повторений, а цикл `while`, когда количество повторений неизвестно:

```
1 for (i = 0; i < 10; i++) {
2     ...
3 }
4
5 while (i < 10) {
6     ...
7 }
```

Разделение функций и блоков кода

Функции и различные по смыслу блоки кода разделяются пустой линией:

```
1 int smthFunc1 () {
2 }
3
4 int smthFunc2 () {
5 }
```

3 Вывод

Был описан мой стиль написания кода, оформленный в редакторе L^AT_EX.

Список литературы

- [1] Осваиваем L^AT_EX за 30 минут (Дата обращения 06.02.22)
<https://habr.com/ru/company/ruvds/blog/574352/>
- [2] Как просто вставить программный код в L^AT_EX без боли и страданий?
(Дата обращения 06.02.22)
<http://mydebianblog.blogspot.com/2012/12/latex.html>
- [3] 90 рекомендаций по стилю написания программ на C++
(Дата обращения 06.02.22)
<https://habr.com/ru/post/172091/>