

Code Style

Степан Митрохин, ИА-032

February 2022

1 Introduction

Создание Код-Стайла для языка C++

2 C++

Currently, the C++14 version is used [1].

Code Style for C++

Пробелы и отступы

Между каждым оператором и операндом рекомендовано ставить пробел, так намного лучше читается код, меньше шансов не заметить какой то символ, который может в противном случае слиться, например с названием переменной

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

Фигурные скобки

Правило постановки фигурных скобок рассмотрим на блоке "if". Фигурная скобка открывается на той же строчке, где написано само условие, открывать ее с новой строчки нет необходимости, это займет лишнюю строчку, так как после открытой фигурной скобки, всё, что находится до ее закрытия сдвигается вправо на один "Tab" а вот закрывать ее нужно на новой строке на одной оси с началом блока, как показано на примере. Это помогает читаемости кода.

```
if ( a == b ) {  
    foo();  
}
```

Пустая строка:

Между функциями и группами выражений оставляется пустая строка:

```
void foo() {  
    ...  
}  
  
// пустая линия  
void bar() {  
    ...  
}
```

Названия и переменные

Называйте переменные, функции и т. д. описательными именами, чтобы можно было по названию понять, что делает эта функция, что значит эта переменная. Каждое объявление стоит выполнять на новой строке. Все это существенно облегчит работу.

Переменные и функции принято записывать в "camelCase" стиле, когда каждое слово в названии пишется слитно и с заглавной буквы, а первое слово пишется с прописной.:

```
int firstName;  
void homeworkScore();
```

Перечисления должны иметь общую для типа часть:

```
Color_One,   Color_Two,   Color_Three, ...
```

Классы называются описательными именами в следующем регистре:

```
class MyClass { ...
```

Константы называются описательными именами в следующем регистре:

```
const int VOTING_AGE = 18;
```

Использование return:

Когда требуется вернуть значение из функции, используйте значение return:

```
int max(int a, int b) {  
    if (a > b) {  
        return a;  
    } else {  
        return b;  
    }  
}
```

Блоки for, while, if, else

Цикл for используется, когда вы знаете количество повторений, а цикл while, когда количество повторений неизвестно:

Повторяет 'size' раз

```
for (int i = 0; i < size; i++) {  
    ...  
}
```

Повторяет, пока больше не будет строк

```
string str;  
while (input » str) {  
    ...  
}
```

Фигурные скобки всегда ставятся при использовании if, else, while, for:

```
if (size == 0) {  
    return;  
} else {  
    for (int i = 0; i < 10; i++) {  
        cout « "ok" « endl;  
    }  
}
```

Если есть выражение if / else, которое возвращает логическое значение, то возвращаются результаты теста напрямую:[2]

```
return score1 == score2;
```

Проверяйте значения логического типа, не используя == или != с true или false:

```
if (x) {  
    ...  
} else {  
    ...  
}
```

Блоки switch, try-catch

Эти блоки, в принципе, оформляются по правилам, изложенным выше. Но для примера напишу как они должны выглядеть.[2]

```
switch (value)  
{  
    case ABC:  
    {  
        ...  
        break;  
    }  
  
    case DEF:  
    {  
        ...  
        // fallthrough  
    }  
  
    default:  
    {  
        ...  
        break;  
    }  
}
```

```
try
{
    ...
}
catch (const Exception& exception)
{
    ...
}
```

Заголовочные файлы

Всегда размещайте объявления классов и их частей, объявления функций в собственные файлы, ClassName.h, functionName.h.

Директивы include группировать по смыслу, в частности более низкоуровневые располагать выше остальных.

Заголовочные файлы должны включать include-защиту. #pragma once [3].

3 Вывод

В ходе проведения данной работы мной был собран основной перечень правил код-стайла для языка C++, для удобства написания и чтения разработанных программ.

Список литературы

- [1] ISO/IEC 14882 Programming languages - C++. [03.02.2022]

- [2] <https://gist.github.com/azoyan/b545f7b926f1f7fb40f8c285e3f5c545>
[03.02.2022]

- [3] <https://tproger.ru/translations/stanford-cpp-style-guide/>
[03.02.2022]