

# Соглашение по написанию программного кода на языках C/C++/Java

## 1 Стили использования регистра букв

1. Pascal case – первая буква идентификатора заглавная, все последующие первые буквы слов тоже заглавные. Примеры: **T**ree, **H**ash**S**et, **G**et**D**ata**A**nd**C**onvert;
2. Camel case – первая буква идентификатора строчная, все последующие первые буквы слов заглавные. Примеры: tree, **h**ash**S**et, **g**et**D**ata**A**nd**C**onvert;
3. Snake case – все буквы идентификатора строчные, слова разделяются знаком нижнего подчеркивания. Примеры: tree, hash\_set, get\_data\_and\_convert;
4. Screaming snake case = snake case, но все буквы в верхнем регистре. Примеры: TREE, HASH\_SET, GET\_DATA\_AND\_CONVERT.

## 2 Файлы

Как отдельные файлы, так и файлы проекта должны именоваться в соответствии со стилем **snake case**.

### Дополнительно для C++, Java:

Кроме того, в каждом файле должна содержаться только одна структура данных (класс, структура, перечисление). Исключение можно сделать только для простейших вспомогательных объявлений, а также при выполнении учебных задач на занятии для экономии времени. Сам файл исходного именуется в соответствии со структурой данных, которая в нем описан.

## 3 Переменные и поля классов

Переменные и поля именуются в соответствии со стилем **camel case**. Их имена должны быть разумными: в циклах допустимы переменные-счетчики i, j, k; в остальных случаях смысл переменной должен быть понятен по её названию.

## 4 Функции и методы

Функции и методы именуются в соответствии со стилем **camel case**. Их параметры также именуются в соответствии со стилем **camel case**. Наименования функций и параметров также должны быть понятными и разумными.

## 5 Константы

Константы именуются в соответствии со стилем **screaming snake case**.

## 6 Перечисления

Перечисления именуются в соответствии со стилем **pascal case**. Элементы перечислений именуются в соответствии со стилем **screaming snake case**.

## 7 Классы и интерфейсы

Классы и интерфейсы именуются в соответствии со стилем **pascal case**.

## Порядок объявления элементов класса

1. Объявление констант;
2. Объявление полей класса;
3. Объявление конструкторов класса (первым идет конструктор по умолчанию, если он есть);
4. Объявление методов класса (сначала геттеров и сеттеров, затем остальных методов).

## 8 Отступы

В качестве отступов используется либо табуляция (предпочтительнее), либо 4 пробела, в каждом проекте использовать только выбранный стиль отступов.

## 9 Положение фигурных скобок

Допустимые варианты расположения фигурных скобок в тексте программы представлены в таблице:

Рациональный стиль	Стиль Алмена
<pre>int main() {     if(...) {         ...     } else {         ...     } }</pre>	<pre>int main() {     if(...)     {         ...     }     else     {         ...     } }</pre>

Рациональный стиль, как правило, применяется при программировании на Java (см. java code conventions), стиль Алмена чаще встречается при программировании на C/C++. В любом случае, при программировании на C/C++ можете выбрать любой стиль, но нельзя использовать разные стили в одном проекте.

## 10 Сборка

При сдаче заданий на языках C/C++ в случае наличия нескольких файлов с исходным кодом также должен быть предоставлен makefile (в случае отправки проекта Qt Creator конфигурационный файл создается автоматически).

## 11 Дополнительно

Наименования всех идентификаторов должны быть понятными и разумными в данном контексте.

Если строка с исходным кодом занимает значительное место по горизонтали, рекомендуется разбить ее на несколько строк.