

C and C++ Coding Conventions

File Conventions

- Header files: `.h` for C, `.hpp` for C++
- Implementation files: `.c` for C, `.cpp` for C++
- File names are typically `snake_case`
 - Example: `my_header_file.h`, `my_implementation_file.cpp`

Function Conventions

- Function names are typically `snake_case` in C and `camelCase` or `snake_case` in C++
- Function names should be verbs if the function performs an action
- Parentheses should be placed with no spaces between the function name and the parentheses
 - C Example: `void perform_action() { /*...*/ }`
 - C++ Example: `void performAction() { /*...*/ }`

Variable Names

- Variable names are typically `snake_case` in C and `camelCase` or `snake_case` in C++
- Variable names should be nouns
- Avoid single character names except for loop indices
- Avoid using names that could be confused with keywords
 - C Example: `int my_variable = 10;`
 - C++ Example: `int myVariable = 10;`

Other Conventions

- Constants are typically `UPPER_SNAKE_CASE`
 - Example: `const int MY_CONSTANT = 100;`
- Use spaces around operators and after commas to improve readability
 - Example: `int result = a + b;`
- Always use braces `{}` even for single statement `if`, `for`, `while`, etc.
 - Example: `if (condition) { return; }`
- Indent with spaces, not tabs. A common convention is 4 spaces per indentation level
 - Example:

```
if (condition) {  
    performAction();  
}
```

Remember, these are general conventions and can vary based on the specific coding standards of your project or organization.