# CS2308
# Foundations of CS II
## Style Guide

---

**Code should be easily readable and should expose the intention of the author. The guidelines below should be used when working on Coding Projects and Weekly Assignments.**

1. **Keep every line of code shorter than 80 characters.**
2. **Give variables meaningful names. The name should make the use of the variable obvious.**
3. **Indent every block of code by one tab or four spaces from its parent statement.**
4. **Comments should be included to explain every "complicated" action.**
5. **Every file should include an author, collaborator, and date comment.**
6. **Functions should have one "purpose" and the use of the function should be obvious without reading its definition. The following comments should be included with every function other than *main*:**
    a) **Name- a plain name for the function not in snake or camel case.**
    b) **Parameters- the datatype, name, and use of every parameter to the function.**
    c) **Returns- the value returned by the function or "None" if the function is void.**
    d) **Side Effects- any change the functions makes to the program (e.g. updating a global variable, updating a by-reference parameter, or overwriting a file).**
7. **Avoid code duplication. The same statement should not appear in multiple branches of a program.**
8. **Variable and function names can be in snake_case or camelCase, but pick one and be consistent.**

# Example of Confusing Code:

```
//no authorship comments
#include<iostream>
#include<fstream>

using namespace std;

//missing function comments
float foo(float a){
    ofstream f("out.txt");
    //function has two "purposes"
    f << a+273.15;
    return 32.0 + (a * (9.0/5.0));
}



int main(){
    //meaningless variable name
    float t = 20;
    //missing comment. what does this loop do?
    for(int i = 0; i < 7; i++){
    //confusing indentation
    if(i%2!=1){cout << foo(t) << endl;t++;}
    //duplicated code
    else t++;}

    return 0;
}
```