Assignment 7.b (group part)

Group Members:

Donghao Lin

Christopher Matian

Audrey Young

Benjamin Jones

Our group decided that Chris's implementation of assignment 7b was the best. While all our assignments were similar and produced correct output, we chose Chris' assignment because it had a concise implementation of the standard deviation function, avoided namespace pollution, and had good comments throughout explaining his code. Chris' standard deviation function is implemented correctly and doesn't have any redundancy of mathematical operations. The variables have descriptive names, and the code is easy to follow and self-documenting. What separated Chris's implementation was his precise use of importing from namespace. He avoided "using namespace std" and specified the specific names with predefined meanings (ex. using std::string, using std::cout, etc..) that he needed for each file. In addition, Chris also had the most thoroughly documented code, clearly explaining his thought process with inline comments.

One of the improvements that could be made to this implementation is removing some redundancy in the headers. Since Person.hpp already has #include <string> and using std::string, we can remove these from Person.cpp and stdDev.cpp because both those files already #include "Person.hpp". We can also remove #include <iostream>, using std::cout, and using std::endl from Person.cpp since the Person class does not utilize input or output.

A second modification would be to replace the exponent value from an integer (2) to a double (2.0). It was noted that the pow function expects a double type definition as a parameter and that it would be wise to avoid any unintended behavior (integer division).

Thirdly, the structure of the class in *Person.hpp* can also be altered by moving the public class up in hierarchy, and placing the private data elements at the bottom. For all intents and purposes the code functioned without any issues, however, in a working professional environment, it helps to place the innards (methods, constructors, etc) of the class at the top of the list for the sake of having a better frame of reference when viewing code.

Overall, each member of the group shared code that was similar in nature and structure, and the end result returned the expected output. Ultimately, what it came down to was which implementation had better documentation of code and namespace formatting.

Noted Differences:

	Person.hpp	Person.cpp	stdDev.cpp	Other thoughts
Ben:		• Doesn't include <iostream> and <string> or "using namespace std"	• Uses <math.h> • Doesn't include <iostream>, "Person.hpp," "using namespace std" • No main function	• Lots of comments
Chris:	• Includes <iostream> • Starts with different order		• Includes <string>	• Lots of comments
Donghao:	• Includes <iostream>	• Includes a default constructor	• Sets size=N	• Lack of comments and explanations in code.
Audrey:	• Names the values the constructor passes	• Doesn't include <string>	• Uses "typedef int arrayType[]" • Doesn't comment main function out	• Doesn't include side comments/ explanations