Grading Criteria

Correctness- 70%
This will include:
• Does your program accomplish the core required homework task?
• Have you checked for edge cases? I.e. Is it comprehensive?
• Does your program catch errors? I.e. Is it robust?
• Note: If your program does not compile, it will result in a 50% loss of correctness points. Please make sure that your solution compiles when you submit it.

Style- 30%
This will include:
• Meaningful comments - this does not mean you should comment every line of code. More comments do not necessarily mean a better grade; excessive commenting of obvious code can make a program equally unreadable. However you should get in the habit of including range comments at the beginning of functions to describe their purpose/arguments/output and inline comments as necessary to aid readability and clarify any non-obvious sequences of code.
• Code Readability - do your variable and function names indicate their purpose? 'x' is almost never a good name for a variable as it is unclear as to what is stored in x. A better name would be 'avgAge', as someone reading the code would immediately know what was stored in the variable.
• Consistency - once you choose a style of code, please stick with it. This applies to names (myVariableName vs my_variable_name), spaces ( x = 2, that is, a blank space around operators,), returns (if you have two lines between functions at the beginning of your code, you should not then separate functions later on in your code using four lines), etc. You will be graded on consistency, not which style you choose, but it may be worthwhile to practice using some of the industry standard coding styles (i.e. PEP8 for Python or Code Conventions by Oracle for Java).
• Elegance - does your algorithm include a lot of unnecessary work or brute force calculations? Is it patched together haphazardly without clear organization or purpose? Do you use excessive memory or will your algorithm be exceedingly slow?