AP Computer Science Principles, Creating a Program of your own

Your program must have at least one algorithm that is made up of two algorithms. This means that the two algorithms within this one algorithm somehow work together to be considered one algorithm, yet at least one of the two algorithms can also work independently. The whole algorithm must also contribute to some larger purpose of the program and so should not be the entire, or almost the entire, program itself.

Your program must have at least one abstraction, most probably a list or a function, that clearly manages the complexity of the program.

Your program must be unique, although another student can help you. You must have done most of it independently though and be able to explain it fully when pressed. If I help you, it cannot be considered toward the actual Create Task.

Due date: December 15th, 2017

Note: You may enhance your existing project, but the addition in and of itself must include the requirements above.

Algorithm: Uses math operators and/or logic, cannot consist solely of calls to existing library functions, cannot consist of a single instruction.

On paper, in the code, or at the bottom of your code, document your difficulties (as they pop up) and how they are resolved.

Always put #comments into your code to clarify what the code is doing and how the code is working.

Points of clarification:
1) “print” in and of itself will not be considered an instruction
2) An if/else statement (since the user only takes one of the paths) will only be considered a multi-instruction algorithm if there is more than one instruction in each of the paths.
3) Any instruction that can be iterated using a loop is not in itself a multi-instruction algorithm.
4) “almost the entire” (stated above) is considered more than 60%

IN OTHER WORDS, YOU ARE NOT AT 100% UNLESS YOU HAVE AN ABSTRACTION THAT CLEARLY MANAGES THE COMPLEXITY OF YOUR CODE AS WELL AS AT LEAST ONE ALGORITHM THAT ITSELF IS MADE OF AT LEAST TWO MULTI-INSTRUCTION ALGORITHMS.