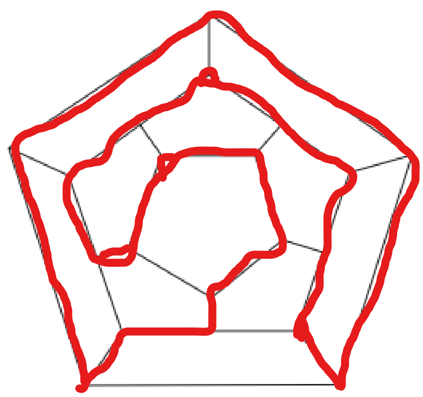
Homework 1 CSCE 350

(20) 1.1.4, modified: Design an algorithm for computing $\lfloor\sqrt{n}\rfloor$ for any positive integer *n*. Besides assignment and comparison, your algorithm may only use the four basic arithmetical operations. For full credit it needs to be O(log n) or O(b) for full credit, where b is the number of bits in the binary representation. What this means in practice is that a number "around" a million should take twice as long as "about" 1000 and about two thirds as long as one billion; 1000 ~=2^{10}, 1,000,000 ~=2^{20}, 1,000,000,000 ~2^{30}. You might think of algorithms you've (probably) learned that run in O(log n) time for *inspiration*.

(10) 1.3.5: (just draw on the graph and color/draw on it) Icosian Game A century after Euler's discovery (see Problem 4), another famous puzzle—this one invented by the renowned Irish mathematician Sir William Hamilton (1805–1865)—was presented to the world under the name of the Icosian Game. The game's board was a circular wooden board on which the following graph was carved:



Find a Hamiltonian circuit—a path that visits all the graph's vertices exactly once before *returning* to the starting vertex—for this graph.

(20) 1.3.9: Design an algorithm for the following problem: "Given a set of n points in the Cartesian plane, determine whether all of them lie on the same circumference."

- "lie on the same circumference" means "there a exists a circle that contains all the points"
- you don't have to worry about rounding errors with floating points numbers, i.e. you may assume that there is no error to point locations from using a fixed number of bits to represent them (or from measurement error).
- You should in a comment for pseudocode, explain briefly your approach.
- (Side note: some of you will have had a geometry teacher tell you a fact that makes this relatively easy. Try to develop an understanding for *why* the fact is true to write the (pseudo)code.)