

Jacob Hamilton      725009698  
Jake7054      jacobhamilton@tamu.edu

Please list all sources in the table below including web pages which you used to solve or implement the current homework. If you fail to cite sources you can get a lower number of points or even zero. According to the University Regulations, Section 42, scholastic dishonesty are including: acquiring answers from any unauthorized source, working with another person when not specifically permitted, observing the work of other students during any exam, providing answers when not specifically authorized to do so, informing any person of the contents of an exam prior to the exam, and failing to credit sources used. Disciplinary actions range from grade penalties to expulsion, read more: Aggie Honor System Office.

Resources used: cplusplus.com

I certify that I have listed all the sources that I used to develop the solutions/codes to the submitted work.  
*"On my honor as an Aggie, I have neither given nor received any unauthorized help on this academic work"*

Jacob Hamilton      4/28/19

Description of Data Structures: Used an Adjacency List. It is a vector of a list of all vertices and what they connect to.

Conditions for drawing one-stroke pictures: There must be a way to draw the graph without overlapping or going over one path more than once.

Description of Algorithm and running time: The program is a vector of lists of all the vertices. It iterates through the vector and runs `search()` with each list in the vector. The search function keeps a stack that holds the path it takes. It will recursively call the next ones in the list and if it does not work it will `pop()` the last one off the stack. The running time is  $O(n!)$  in the worst case. In the best it would be  $O(n)$ .

Evidence of testing: On the next page

Graph 1:

1->2->3  
2->1->3->4->5  
3->2->1->5  
4->2->5  
5->4->2->3

Has a path

Starting vertex: 3

3->5->4->2->1->3->2->5

Graph 2:

1->2->3  
2->1->3->4->5  
3->1->2  
4->2->5  
5->2->4

Has a path

Starting vertex: 1

1->2->4->5->2->3->1

Graph 3:

1->2->3->4  
2->1->5->6  
3->1->4->7  
4->1->3->9  
5->2->6  
6->2->5->10  
7->3->8->11  
8->7->11  
9->4->10->12  
10->6->9->12  
11->7->8->12  
12->9->10->11

Does not have a path

Graph 4:

1->2->3  
2->1->4->7  
3->1->4->5->6  
4->2->3->5->6  
5->3->4->6  
6->3->4->5->7  
7->2->6

Has a path

Starting vertex: 2

2->1->3->4->2->7->6->4->5->3->6->5

Graph 5:

1->2->3  
2->1->3->4->5  
3->2->1  
4->2->5  
5->4->2

Has a path

Starting vertex: 1

1->2->4->5->2->3->1

Graph 6:

1->2->3->4  
2->1->3->4->5  
3->1->2->5  
4->1->2->5  
5->2->3->4

Does not have a path