

OS project - Graph algorithms, Strategy/Factory, Client-Server, Threads, Active Object, Thread pool (Leader-Follower) and Valgrind

1. Create a Graph Data structure
2. Implement an Algorithm to find Euler circle (Euler circuit) on the graph or prove one does not exist.
3. Generate a Random graph and run the algorithm on it. Receive parameters (number of edges, number of vertices, random seed) using argc,argv with getopt(3) - same as ex.1
4. Provide code coverage reports, gprof, Valgrind/memcheck report and Valgrind callGraph for your code,
5. The following code has some problems. Run Valgrind/memcheck on it. And report the errors (10 pts)
6. Implement a server that receives requests for Euler circle on graphs
7. We may now be interested in other problems on the graphs
 - a. Add 4 graph algorithms of your choice from the following list.
 - i. Finding max clique
 - ii. Finding number of cliques
 - iii. Finding MST weight
 - iv. Finding Hamilton circuit
 - v. Finding SCC on the graph
 - vi. Finding max flow between node 0 (source) and node (n-1)
 - vii. Finding node disjoint minimal path cover on the graph
 - viii. Finding the maximal antichain (Dilworth's Algorithm)
 - b. Implement a server that gets an algorithm request and provides the answer.
 - c. The server and algorithms should be packed using the Factory and Strategy patterns.
8. Multithreading - LF
 - a. The server should implement multiple threads
 - b. The server will receive a graph (or request to generate random graph) from the clients and provide answers to all 4 algorithms implemented in section 7 in a LF fashion
9. Multithreading - Pipeline
 - a. Same as section 8 but the server should include 4 (or more) active objects and provide the response using the pipeline pattern. (you may add more active objects to receive the request and send the response, but atleast 4 (for the four algorithms) are required).
10. Provide Valgrind analysis (memcheck, helgrind, cg)
11. Prove code coverage of all code