#Exercise 4 Backtracking algorithms

Uladzimir Ivashka, 150281

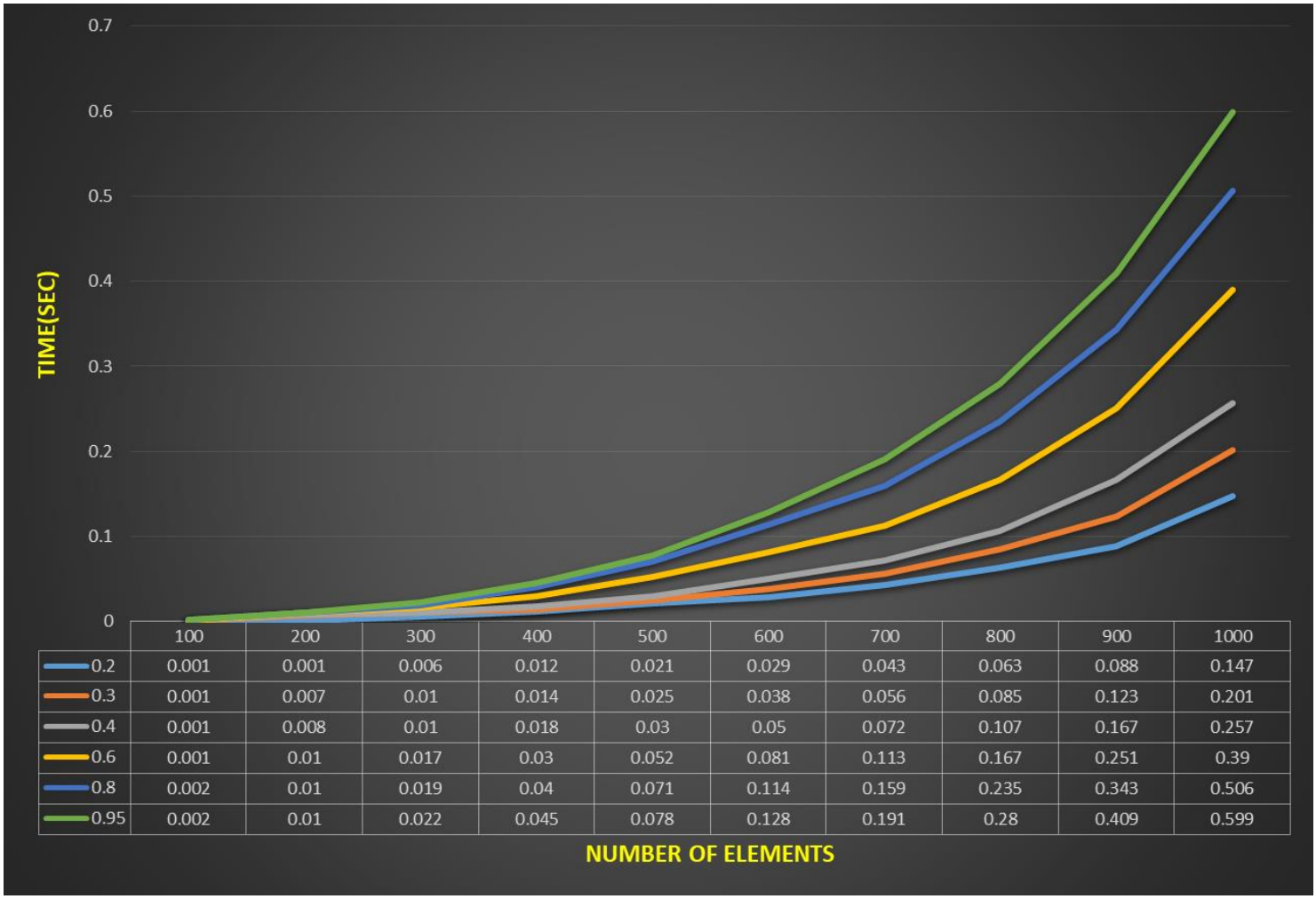
Sofya Aksenyuk, 150284

Remark:

Given results of all the below experiments are average of 10 runs.

Exercise I.

Comparison of Euler cycle search time in undirected randomly generated connected graph with different edge saturations (0.2, 0.3, 0.4, 0.6, 0.8, 0.95):



	100	200	300	400	500	600	700	800	900	1000
0.2	0.001	0.001	0.006	0.012	0.021	0.029	0.043	0.063	0.088	0.147
0.3	0.001	0.007	0.01	0.014	0.025	0.038	0.056	0.085	0.123	0.201
0.4	0.001	0.008	0.01	0.018	0.03	0.05	0.072	0.107	0.167	0.257
0.6	0.001	0.01	0.017	0.03	0.052	0.081	0.113	0.167	0.251	0.39
0.8	0.002	0.01	0.019	0.04	0.071	0.114	0.159	0.235	0.343	0.506
0.95	0.002	0.01	0.022	0.045	0.078	0.128	0.191	0.28	0.409	0.599

Conclusions:

According to the graph above, it can be noticed that it takes more time to find Eulerian cycle in a graph with higher edge saturation. It can be explained by the essence of Eulerian cycle: it needs to check all edges to check whether it exists or not, therefore, more edges – more time to go though all of them.

Hierholzer's algorithm was chosen for the presented computations due to time complexity of O(E) of its work, whereas it is equal to O(V^2) for Fleury's algorithm.

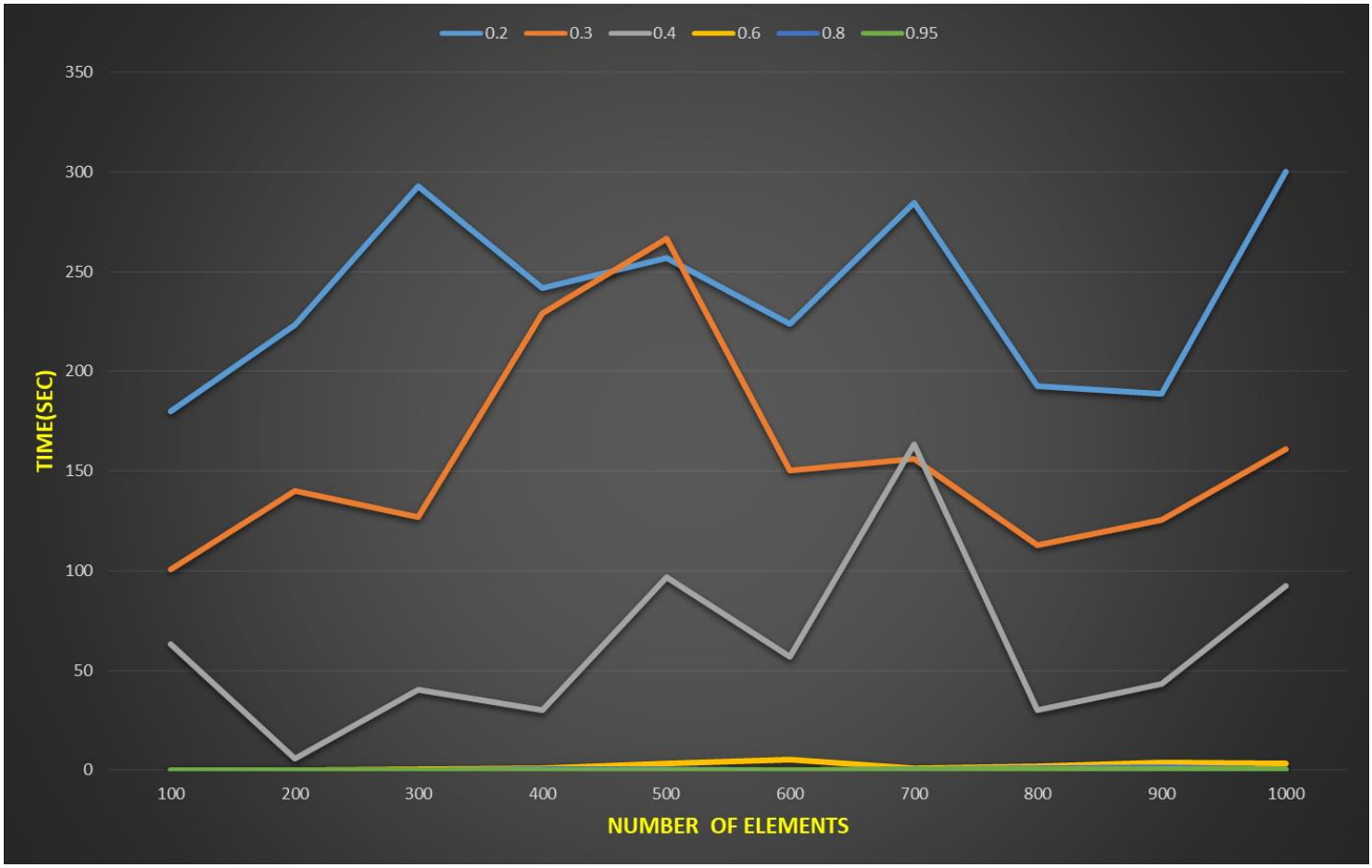
As for graph representation, vertex matrix was chosen because of the need to store an information about the existence of an edge between vertices for further computations of the amount of their degrees. Eulerian cycle is obtained in a connected graph if the degree of each vertex is even or only two of them have odd degrees (they will be considered as start and finish nodes).

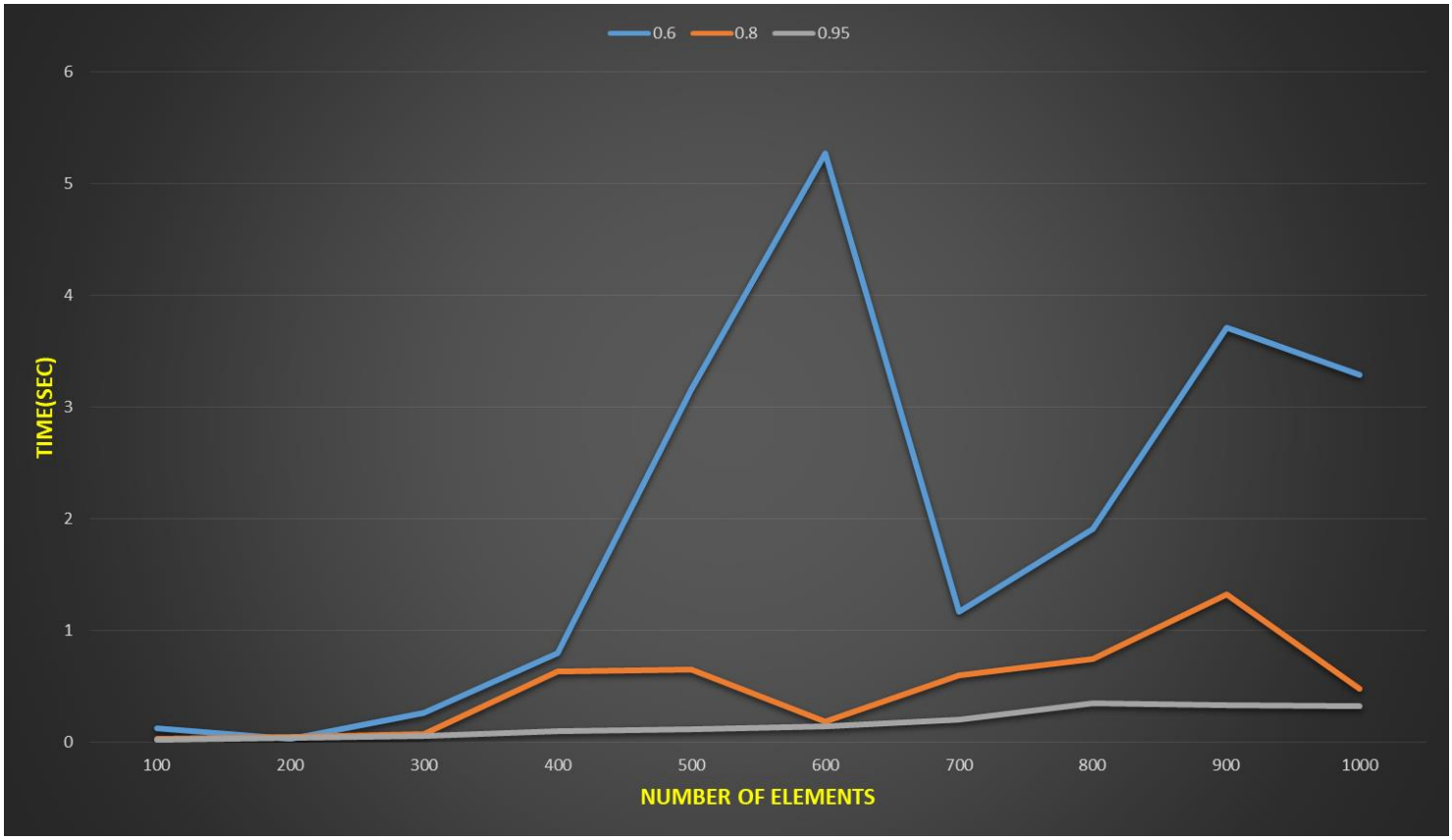
The main advantage of vertex matrix in our case is its time complexity for an element deletion (O(1)), whereas it is O(n) for Incident List and Edge List and O(V+E) for Incident Matrix. Deletion function is needed to delete visited nodes.

Exercise II.

Comparison of Hamiltonian cycle search time in undirected randomly generated connected graph with different edge saturations (0.2, 0.3, 0.4, 0.6, 0.8, 0.95):

Remark: Stopping criterion of 300 seconds was applied.





Conclusions:

Hamiltonian cycle exists if there exists a path that visits each vertex exactly once and returns to the starting point. It can be noticed, that higher edge saturation is – faster it is to find Hamiltonian cycle. It happens because the amount of paths creating Hamiltonian cycle increases, and, therefore, it becomes easier to find such a cycle.

Backtracking recursive algorithm that was used gives us time complexity of O(2^n) which can be explained by its principle: it selects starting node and forms some kind of tree of all the vertices it leads to, calling this recursion from each of these nodes.