Lib name is Graphs.

There are two interfaces in package “ru.topper.graphs.entities” which user has to implement to use the lib.

First interface is Edge. It contains four getter/setter pairs:

1 Strat vertex ID

2 Finish vertex ID

3 Edge weight (optional, if user wants to implement weighted edges) default is 1

3 Is edge directed (optional, if user wants to implement directed graph) default is not directed

And it has clone method to split one edge object for two in case of not directed graph.

Second interface is Vertex. It contains three getter/setter pairs:

1 Vertex ID

2 Vertex edges map

3 Is vertex visited (for traverse algorithms)

There are two implements of the interfaces.

UserEdge. It is universal. It has four constructors in cases object weighted or not, directed or not. And it implements all default methods.

UserVertex. It implements all methods and adds user string variable “name” and its getter/setter.

It implements Comparable by name for some reason too (just for example)

GNet in “ru.topper.graphs” is the lib`s main class, it has vertex map and contains methods:

1 addVertex adds a vertex to the map.

2 addEdge adds an edge to vertex’s edge map (synchronize on the vertex map because we change two vertex in case of undirected graph). This method add the edge clone to start vertex as is, and to finish vertex but swap start and finish in the clone (in case of undirected graph)

3 getVertexes returns vertex map, so we can use it in ours traverse function

4 getPath returns the shortest path between start and finish using Deixtra algorithm

5 getPathLenght returns destination by path, if a vertex in the path is not connected then null

6 depthTraverse(Long start) implements Depth-first traversal algorithm, if we want to start from definite vertex

7 depthTraverse() implements Depth-first traversal algorithm, if start is no matter

### 8 breadthTraverse(Long start) implements Breadth-first traversal algorithm

### All methods synchronized on the vertex map for writing/reading thread safety.

There is class Using as example of using the lib.