

Shortest Path Algorithm Visualization Project

1 Project Brief

In order to better understand some algorithms of the shortest path calculation, and test our algorithm writing ability as well, we determine to use JAVA to write our algorithmic codes to calculate the shortest path we input. Besides, to show the effect of the different algorithms and compare, we take advantage of JAVA Graphical User Interface to achieve the visualization of these algorithms.

We use three algorithms totally, including Dijkstra, Bellman-ford and Spfa. Because of the single origin we set, we have to give up Floyd algorithm which is designed for multiple origins.

2 Technology Introduction

2.1 Algorithms Achievement

2.1.1 Dijkstra

The main feature of Dijkstra algorithm is to extend from the starting point to the outer layer until the end point.

To implement Dijkstra, the node set in Graph is divided into node set S with the shortest path calculation completed and node set T with the uncalculated. Each time, the node v_i with the smallest $v_0 \rightarrow v_i$ from T is selected to be added to S , and the shorter distance between v_0 and the remaining nodes in T is updated through v_i until all nodes in T are added to S . It's greedy that every time it chooses the node closest to the source to join the shortest path node set. (S and T is a set of examples of name)

2.1.2 Bellman-ford

Bellman-ford algorithm's remarkable feature is that it can find a single source shortest path with negative weight graph, and judge whether there is a negative weight loop or a point with a shortest path in the graph.

In the graph, each point holds a value representing the distance from the origin, setting the value of the origin to 0 and the value of the other points to infinity. And then loop from 1 to $n-1$ (n equals the number of points in the graph). Inside the loop, all edges are traversed and relaxation calculations are performed. Traverse all the edges on the way, judging whether there is a loop with negative weight reachable from the source.

2.1.3 SPFA

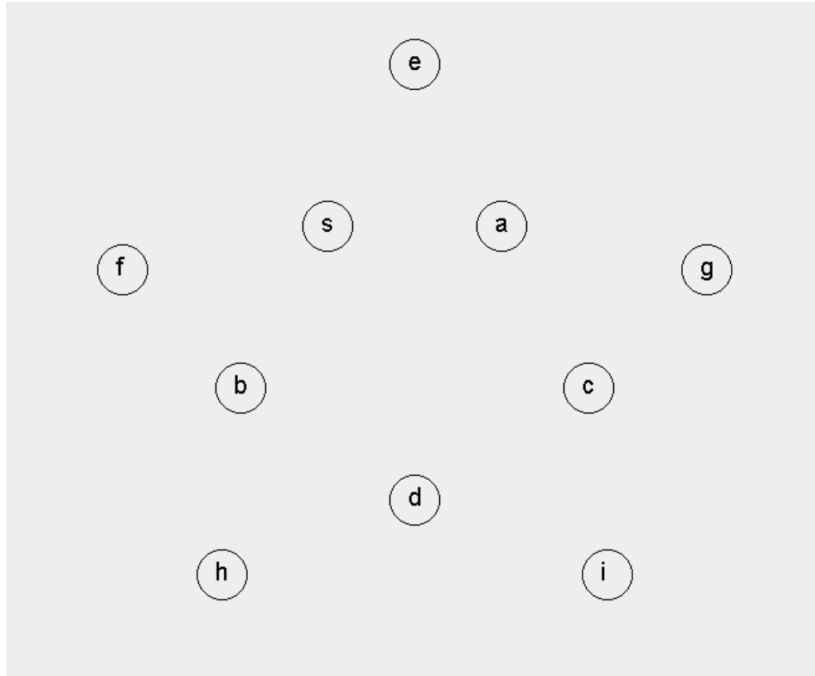
Bellman-Ford algorithm has a high time complexity, because Bellman-Ford needs to recurse for n times, and each recursion needs to scan all edges. In the process of recursion for n times, many judgments are unnecessary, so queue optimization is considered to reduce unnecessary judgments. This algorithm is called SPFA. Maintenance is done using a priority queue, where the source point is initially queued, one vertex is removed from the queue at a time, and all its neighbors are relaxed. If a vertex is successfully relaxed, it is queued, and the process is repeated until the queue is empty.

2.2 Visualization Achievement

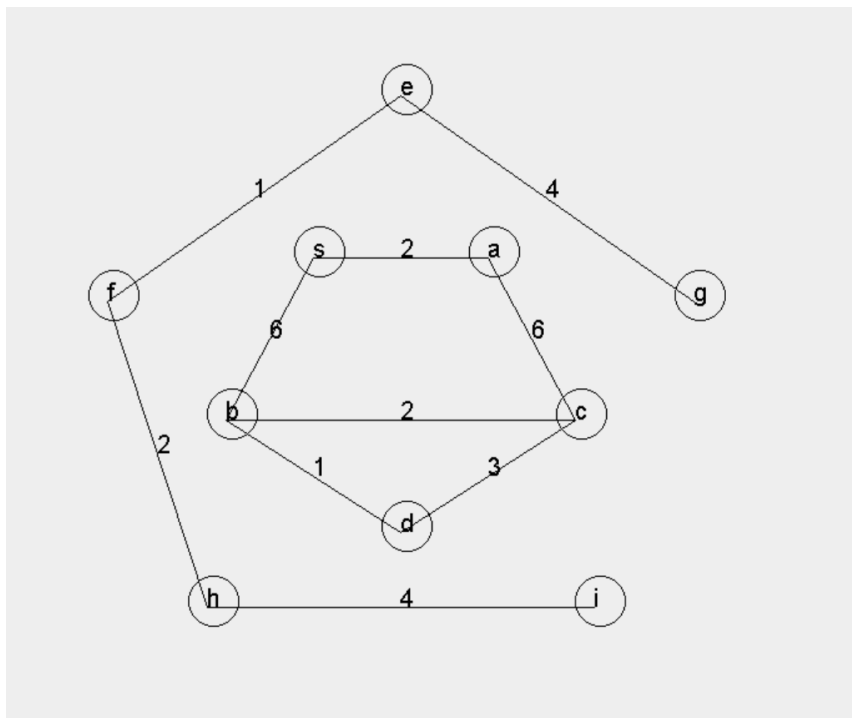
JAVA Graphical User Interface

2.2.1 Graph Outputting

Use *Graphics* from JAVA to draw the graph we want to show. We have positioned the coordinates of the vertexes, edges and their labels already.



graph 1 The initialization coordinates of all the vertexes



graph 2 A test of graph outputting

2.2.2 Animation Demonstration

We store a set of time steps and utilize Thread in JAVA to show each step.

3 Demand Analysis

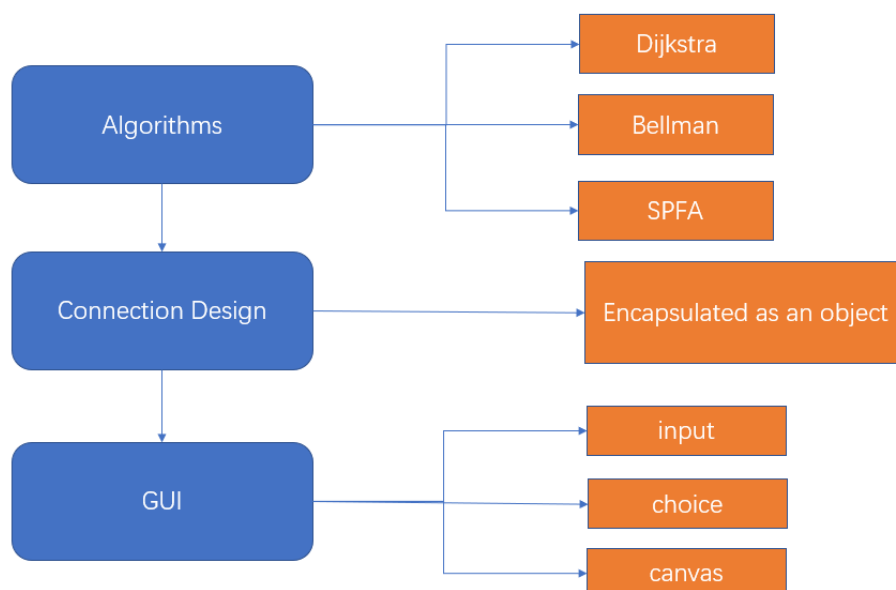
Our project pursues meeting the following requirements:

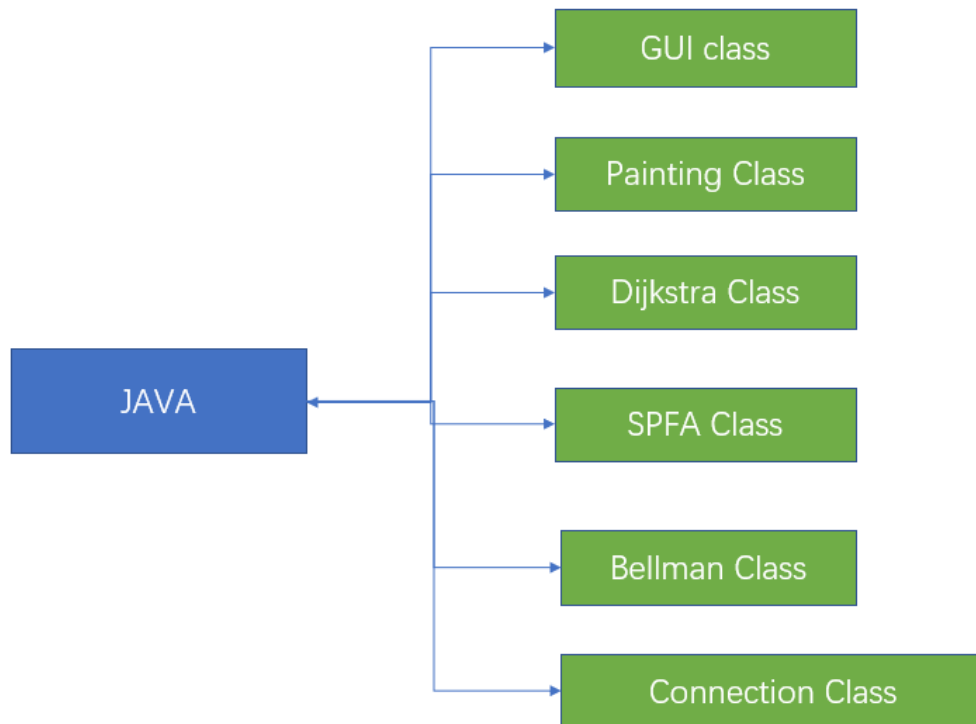
- (1) Achieve the different algorithms to calculate the shortest distance of the graph we input.
- (2) Make a comparison between different algorithms to calculate the shortest path, including their run time and run memory.
- (3) Realize the visualization of the calculation process of various algorithms.
- (4) Our project can be implemented in the practical problem solving.

4 General Design

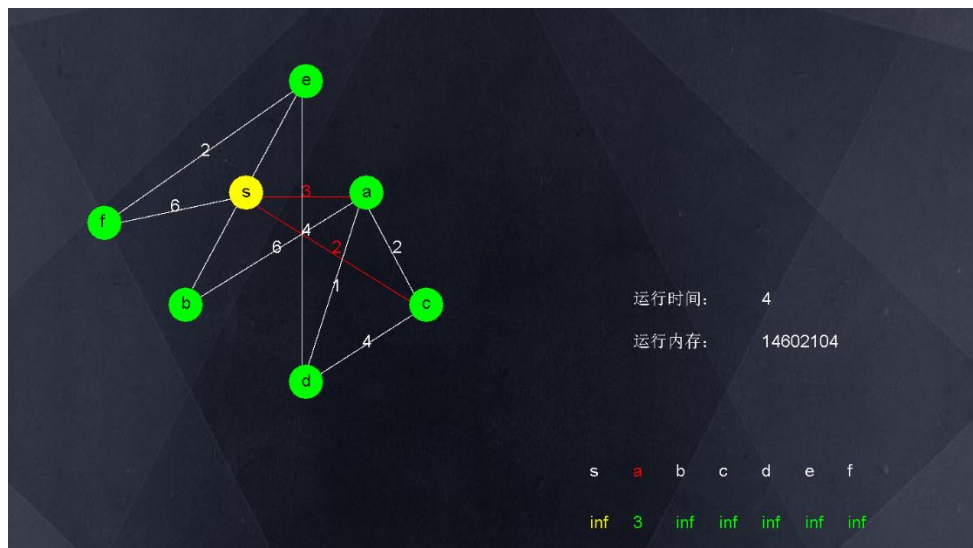
We utilize an interface to store all the data in order to better use for interaction, a class named PuttingGraph to achieve the graph transformation and three classes which capsule these algorithms.

5 Function realization





6 Program Testing



7 Summary and Prospect

The study of shortest path algorithm is a hot topic in computer science. It has not only important theoretical significance, but also important practical value. Shortest path problem has a wide range of applications, such as in the transportation system, emergency rescue system, electronic navigation system and other research fields. The shortest path problem can be extended to the fastest path problem, the lowest cost problem, etc., but their core algorithm is the shortest path algorithm.