

Homework 5 Solving Shortest Path Problem Using POSIX Threads

[Computer Architecture I ShanghaiTech University](#)
[HW4](#) HW5

Shortest Path Problem

In graph theory, the [shortest path problem](#) ([Chinese](#)) is the problem of finding a path between two vertices (or nodes) in a graph such that the sum of the weights of its constituent edges is minimized.

Problem Setup

In this homework, you will try to design a parallel algorithm and implement it using [POSIX Threads](#) ([Chinese](#)). There are many serial algorithms that can solve this problem very efficiently. However, not all of them are good candidates for parallelization.

The practical usage of this algorithm is on distributed systems, where it is difficult to compute the shortest distance on one node and broadcast the result to all other nodes.

You will be given a directed graph with positive edges. Your objective is to find the shortest path between two given vertices (source and target).

Obtain the files

Fork the [framework](#) to your own repository. Please **DO NOT** fork to CS110 or CS110_Projects group, fork to your own namespace. After forking, set it to private immediately. This can be done via `Setting->General->Visibility, project features, permissions`.

Everyone should keep a personal Git repository to record your development. Please spend some time learning how to use Git if you are still not familiar with it.

Input

The maximum available threads on the host machine will be passed as a command-line argument (e.g. `./dist 4`).

The first line of input contains 3 integers n , s and t

- n means the number of vertices.
- s means the index of source vertex.
- t means the index of target vertex.

The next n lines represent an [adjacency matrix](#) ([Chinese](#)) A .

$A[i][j]$ represents the weight of the edge from vertex i to j .

$A[i][j]$ is guaranteed to be non-negative. If $A[i][j] = 0$, there is no edge between i and j .

Sample input

```
3 0 2
0 1 0
0 0 1
0 0 0
```

Output

The first line of output is the value of the shortest path.

The following lines are the shortest path, start from the source vertex. One vertex per line. For each input, it is guaranteed to have only one shortest path.

Sample output

```
2
0
1
2
```

Requirement

- You are only allowed to include header files from the [C standard library](#) ([Chinese](#)) and `pthread.h`.
- No memory leaks are allowed.
- The number of comments should be at least 25% of the non-blank lines. The comments should be valid and in English - failure to comply may lead to a score of 0 for this HW.
- You must use multiple threads to solve this problem. It is not allowed to use multiple threads to handle input or output and use **ONLY** one thread to get the shortest path.

Submission

Submit `dist.c` to Autolab.

Schwertfeger, Sören <[soerensch AT shanghaitech.edu.cn](mailto:soerensch@shanghaitech.edu.cn)>

Chundong Wang <[cd_wang AT outlook.com](mailto:cd_wang@outlook.com)>

Last modified: 2020-04-13