C++ Graph Library Tutorial

April 24, 2018

1 Introduction

2 Start

make your node type

make your edge type

use your node and edge to define graph comform to your use senario

now you have your own graph, you can see it clearly

3 Use our algorithm

path_exists findpath

4 Define your new algorithm

structure edge used:

struct edge { node handle to; E info; }

note: here the handle is used for user, not the real index stored in adjacency matrix or adjacency list.

The iterator of out edges of a specific node is provided, and the range use of it has been impletmented. You can use it like:

for (const auto& e : g.out(h))

- argument: g: your graph; h: the handle of a specific node; e: the out edge of this code

If you have your path consists of (node_handle), you can use the struct path to store this path, and simply call path.print_path() to get the related information on this path (node, edge, etc)