Graph Report

This is a report about our implementation of graph

Haoyu Wang  
Bachelor of EngineeringComputer EngineeringMontreal, Canada  
victorwanghy@gmail.com

Zifen Huang  
Bachelor of EngineeringElectrical EngineeringMontreal, Canada  
huangzifenbb@gmail.com

*Abstract*—We deeply studied a kind of data structure - graph. A graph is an abstract data type that is meant to implement the undirected graph and directed graph concepts from the field of graph theory within mathematics. A graph data structure consists of a finite (and possibly mutable) set of vertices (also called nodes or points), together with a set of unordered pairs of these vertices for an undirected graph or a set of ordered pairs for a directed graph. These pairs are known as edges (also called links or lines), and for a directed graph are also known as edges but also sometimes arrows or arcs. The vertices may be part of the graph structure, or may be external entities represented by integer indices or references.We use C++ language’s template class to implement the data structure and complete the corresponding test

Keywords—graph, directed, undirected, c++, template

# Introduction

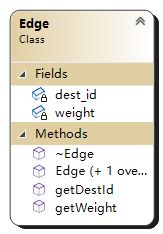
A template is a simple and yet very powerful tool in C++. The simple idea is to pass data type as a parameter so that we don’t need to write the same code for different data types. For example, a software company may need sort() for different data types. Rather than writing and maintaining the multiple codes, we can write one sort() and pass data type as a parameter.

Undirected graph and directed graph can be realized through the inheritance and polymorphic of C++. Then the specific classes are encapsulated according to the differences between the two diagrams.

# Design Description

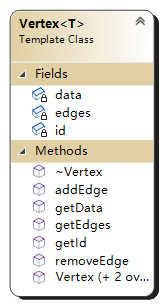
## Class Diagram

This is class diagram of Edge:



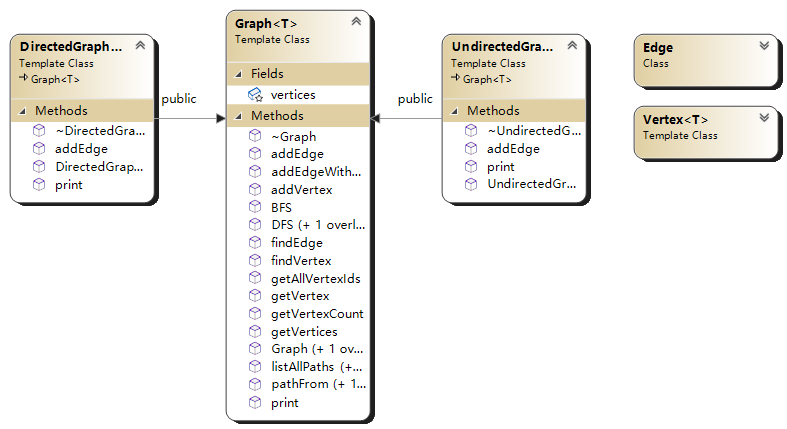
1. Class diagram of Edge.

This is class diagram of Vertex:



1. Class diagram of Vertex<T>.

This is the class diagram about Graph:



1. Class diagram of Graphs.

## Describe about member functions

* Graph’s function addEdge(size\_t start\_id, size\_t end\_id, int weight = 0): This function checks if start\_id and end\_id are valid firstly, then adds an edge from start\_id to end\_id. Specially,If the graph is undirected, two edges will added.
* Graph’s function findVertex(T data): If the value is contained, the function returns the id of the vertex. If the value is not contained, the function returns false.

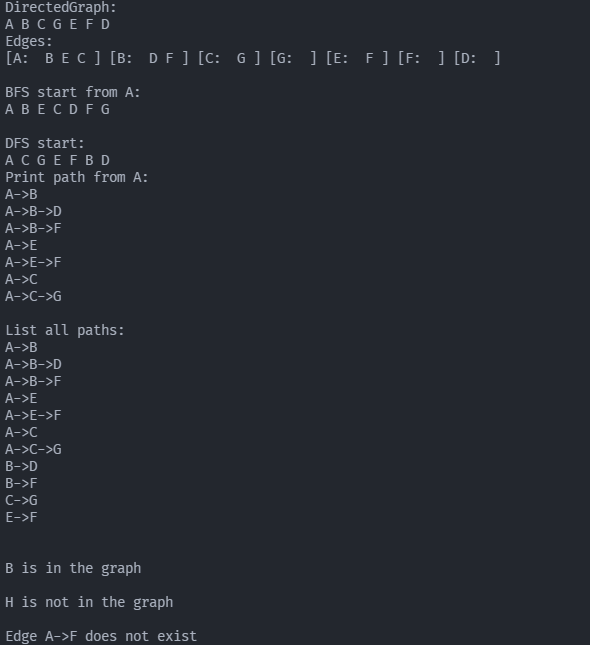
## Usage of techniques

* Inheritance: Graph is the base class to which both directedgraph and undirectedgraph inherit and overload the pure virtual functions.
* Polymorphism: Directedgraph and undirectedgraph are derived classes of graph. Derived class objects can be accessed through the base class pointer, which is the polymorphism of C++.
* Operator Overloading: Graph overloads the operator < < function so that you can easily output the contents contained in the graph.
* Template: Vertex, Graph and it’s derived classes are template classes, which makes the node data of the graph can be of any type.

# Testing Results

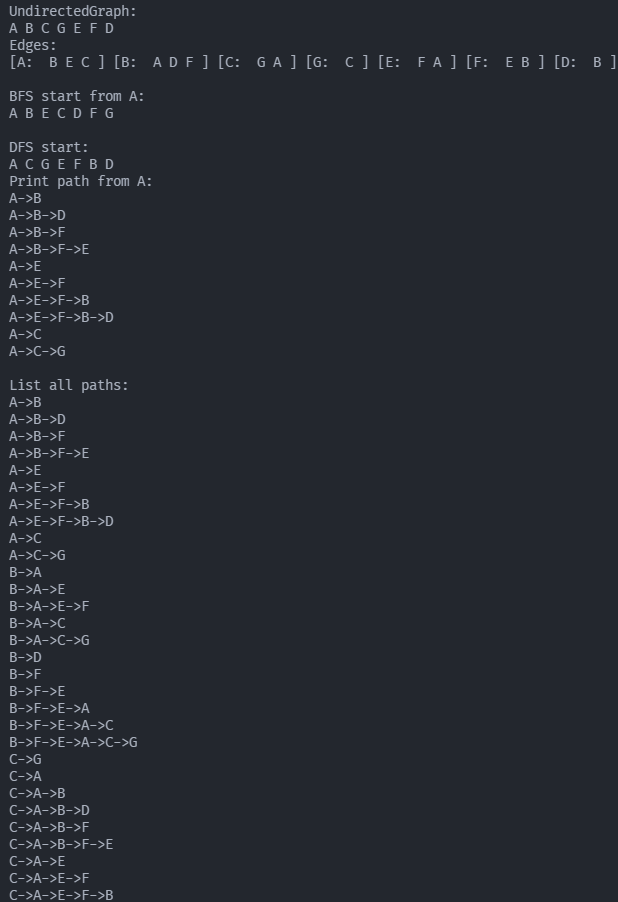
We wrote a driver to test undirected graph and directed graph respectively. The test results are as follow.

## Directed Graph

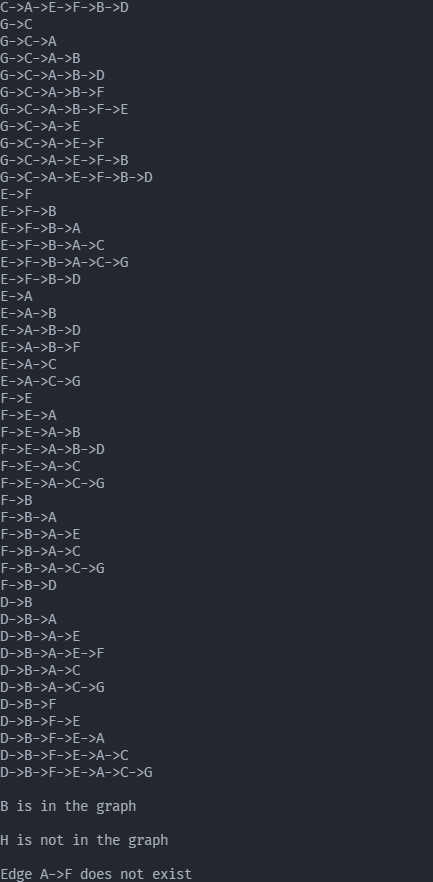


1. Test results of Directed Graph.

## Undirected Graphs



1. Test results of Undirected Graph.



1. Test results of Undirected Graph.

# summary

You can add some summaries and write anything.

##### Acknowledgment

Supplementary thanks.

##### References

Supplement some references

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*
2. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
3. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
4. K. Elissa, “Title of paper if known,” unpublished.
5. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
6. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.