2001年度日本オペレーションズ・リサーチ学会
秋季研究発表会

STLライクな汎用グラフクラスライブラリの紹介

01206340 (株)構造計画研究所 斉藤努 SAITO Tsutomu

1. はじめに

プログラミング言語 C++における汎用のグラフクラスライブラリを紹介する。

STL(Standard Template Library) は C++のテンプレートによる標準のライブラリである。主に、コンテナ(入れ物)と反復子(コンテナの中味を指すもの)からなる。

本稿で紹介する graph クラスは、ノードとアークを入れるコンテナであり、3 つの反復子を提供し、STL と同じように使用できる。

反復子

頂点反復子	頂点を指す反復子
辺反復子	辺を指す反復子
接続反復子	ある頂点に接続している「辺反復子と頂点反復子のペア」を指す反復子

2. 基本方針

- 有向グラフを基本とするが無向グラフとしても利用できる。(使い方はほぼ同じ)
- 頂点(ノード)と辺(アーク)は、任意のクラスを利用できる。
- 一般的な操作が行える。
- 疎なグラフで無駄なメモリを使用しない。
- メモリ管理はSTL同様気にしなくてよい。
- 頂点や辺は反復子で参照する。(頂点反復子は頂点、辺反復子は辺の内容を指す。)
- 頂点の反復子から接続する辺反復子と接続先の頂点反復子のペア(接続反復子)を走査できる。
- 辺の反復子から両端の頂点反復子を参照できる。
- 接続先と接続元の頂点が同じである自己ループや、接続先と接続元が一致する2つ以上の辺の表現はサポートしない。

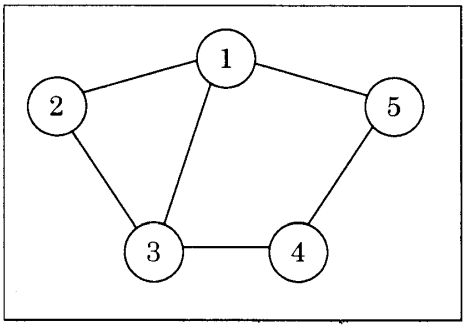
教科書などでよく用いられる隣接行列によるグラフ表現は、大規模で疎なグラフでは非実用的である。graph でのメモリの使用量はO(ノード数)+O(アーク数)である。

3. サンプル

例 1(構築方法と表示)

```
#include "graph.h"
#include <cstdio>
#include <string>
typedef graph<string, int> Graph;

void ShowGraph(Graph& g) {
   Graph::node_iterator    nit, nit2;
   Graph::arc_iterator     ait;
   Graph::connect_iterator cit;
   printf("Graph¥n");
   for (nit=g.nbegin(); nit!=g.nend(); ++nit) {
      printf("%-5s:", nit->c_str());
      for (cit=nit.ibegin(); cit!=nit.iend(); ++cit) {
         nit2 = cit->first; //一度変数に代入
         printf(" %s", nit2->c_str());
      }
      printf("¥n");
   }
   for (ait=g.abegin(); ait!=g.aend(); ++ait)
      printf("  %s - %s¥n", ait.left()->c_str(),
                            ait.right()->c_str());
}
int main() {
   Graph g(false); // 無向グラフ
   vector<Graph::node_iterator> niv;
   const char* nam[] =
            {"One", "Two", "Three", "Four", "Five"};
   int pr[][2] =
          {{0, 1}, {1, 2}, {2, 3}, {3, 4}, {4, 0}, {0, 2}};
   int  i, nn, na;
   nn = sizeof(nam)/sizeof(*nam);
   na = sizeof(pr)/sizeof(*pr);
   for (i=0; i<nn; ++i)
      niv.push_back(g.add_node(nam[i]));
   for (i=0; i<na; ++i)
      g.add_arc(0, niv[pr[i][0]], niv[pr[i][1]]);
   ShowGraph(g);
   g.del_node(niv[2]); // ノード Three を削除
   ShowGraph(g);
   return 0;
}
```




```
出力
  Graph
  One   : Two Three Five
  Two   : One Three
  Three: One Two Four
  Four  : Three Five
  Five  : One Four
    One - Two
    Two - Three
    Three - Four
    Four - Five
    Five - One
    One - Three
  Graph
  One   : Two Five
  Two   : One
  Four  : Five
  Five  : One Four
    One - Two
    Four - Five
    Five - One
```

- ノードは文字列型、アークは整数型。
- add_node でノードを追加する。
- add_arc は、2 つの頂点反復子を引数にとり、そのノードの間にアークを引く。

例 2 では、temp() に距離を入れている。

4. おわりに

本プログラムは、Microsoft Visual C++ 5.0 及び 6.0 と GNU C++ 2.95.2 で稼動確認を行っている。ライブラリの著作権は(株)構造計画研究所が有する。但し、本プログラムを著作権表示を残したまま丸ごと利用するならば自由に利用して頂いて構わない。

ご質問は tsutomu@kke.co.jp に頂きたい。

本ライブラリの開発にあたり社内の方々に助言と公開の承諾を頂いたことに感謝したい。

例 2(最短路探索用メンバ関数)

```
int get_route(const node_iterator& nit1, const
  node_iterator& nit2, list<node_iterator>& l) {
   node_iterator nit; connect_iterator cit;
   list<node_iterator>::iterator lit;
   int    i, cnt = 0;
   int&   len = (*nit1.m_iter)->temp();
   list<node_iterator>   s, t;
   for (nit=nbegin();nit!=nend();++nit)
      (*nit.m_iter)->temp() = -1;
   (*nit2.m_iter)->temp() = 0;
   s.push_back(nit2);
   while (!s.empty()) {
      if (len >= 0) {
         i = len;
         l.clear();
         l.push_back(nit = nit1);
         while (--i >= 0) {
            for (cit=nit.begin();cit!=
                              nit.end();++cit) {
               if ((*cit->first)->temp() == i) {
                  l.push_back(nit = cit->first);
                  break;
               }
            }
         }
         break;
      }
      ++cnt;
      t.clear();
      for (lit=s.begin();lit!=s.end();++lit) {
         node_iterator  noit = *lit;
         for (cit=noit.begin();cit!=
                           noit.end();++cit) {
            int&   k = (*cit->first)->temp();
            if (k < 0) {
               t.push_back(cit->first);
               k = cnt;
            }
         }
      }
      s = t;
   }
   return len;
}
```

5. 参考文献

- 「STL によるコンポーネントデザイン」Ulrich Breymann 著、アスキー出版局、ISBN4-7561-3422-X
- 「STL 標準テンプレートライブラリによる C++プログラミング」デービッド・R・マッサー、アトゥル・サイニ著、アジソン・ウェスレイ発行、ISBN4-7952-9698-7
- 「プログラミング言語 C++第 3 版」Bjarne Stroustrup 著、アジソン・ウェスレイ発行、ISBN4-7561-1895-X