Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Prof. Dr. Ernst W. Mayr
Matthias Baumgart

Sommersemester 2009
Aufgabenblatt 2
4. Mai 2009

Praktikum Diskrete Optimierung

Letzter Abgabetermin: Montag, den 18.05.2009, 14⁰⁰ Uhr

Aufgabe 1 (Minimale Spannbäume (Algorithmus von Kruskal))

Implementieren und animieren Sie den Algorithmus von Kruskal zur Berechnung eines minimalen Spannbaums in einem ungerichteten, zusammenhängenden Graphen G mit positiven, ganzzahligen Kantengewichten. Es soll am Bildschirm sichtbar sein, in welcher Reihenfolge die Kanten von G bearbeitet werden und welche Kanten bereits in den Spannbaum eingefügt wurden. Für den Algorithmus von Kruskal steht Ihnen frei, entweder die Implementierung mittels Priority-Queue oder die Implementierung mit Sortierung der Kanten zu wählen.

Als Eingabe stehen die Graphen `mst1.gw` bis `mst4.gw` auf der Web-Seite zur Verfügung. Die Kantengewichte dieser Graphen sind ganze Zahlen, die im User-Label in String-Form mit abgespeichert sind. Um diese Werte zwecks bequemer Weiterverarbeitung in einem `edge_array` abzulegen, kann eine Schleife der folgenden Form verwendet werden:

```
#include <LEDA/stream.h>
...
leda::edge_array<int> c(g);
edge e;
forall_edges(e,g){
    leda::string s = gw.get_user_label(e);
    leda::string_istream I(s);
    I>>c[e];
    std::cout << c[e] << "\n";
}
```

Aufgabe 2 (Minimale Spannbäume (Algorithmus von Prim))

Implementieren und animieren Sie, analog zur ersten Aufgabe, den Algorithmus von Prim zur Berechnung eines minimalen Spannbaums.