

## Algorithmische Mathematik I

### 5. Programmierübung

Implementieren Sie Kruskals Algorithmus zur Berechnung eines aufspannenden Baumes mit minimalem Gewicht. Ihre Implementierung soll Laufzeit  $O(m \log n)$  haben, wenn  $n$  die Zahl der Knoten und  $m$  die Zahl der Kanten im gegebenen ungerichteten Graphen ist.

Sie können alle Programme und Datenstrukturen, die in der Vorlesung vorgestellt worden sind, direkt verwenden oder geeignet abändern. Insbesondere sollen Sie zum Einlesen des Inputs aus einer Datei den Konstruktor `Weighted_Graph(char const *, bool)` und zum Ausgeben die Funktion `print()` der vorgestellten Klasse `Weighted_Graph` benutzen. Sie dürfen bei Bedarf auch Teile der C++-Standardbibliothek einbinden.

**Einlesen der Daten:** Dem Programm muss beim Aufruf der Name einer Datei übergeben werden. Ein Aufruf hat also die Form

```
<programmname> <dateiname>
```

Eine gültige Datei, die einen gewichteten Graphen beschreibt, hat das folgende Format:

```
Knotenanzahl  
Knoten0a Knoten0b Gewicht0  
Knoten1a Knoten1b Gewicht1  
...
```

In der ersten Zeile steht eine einzelne natürliche Zahl, welche die Anzahl  $n$  der Knoten angibt; Sie können voraussetzen, dass  $n < 2^{32}$  ist. Jede weitere Zeile spezifiziert genau eine Kante. Die beiden ersten Einträge dieser Zeilen sind zwei verschiedene nichtnegative ganze Zahlen, welche die Nummern der Endknoten der Kante sind. Dabei nehmen wir an, dass die Knoten von 0 bis  $n - 1$  durchnummeriert sind. Die Reihenfolge der beiden Knotennummern in einer Zeile spielt keine Rolle, und die Reihenfolge der Kanten kann beliebig sein. Der dritte Eintrag in den Zeilen, die Kanten beschreiben, gibt das Gewicht der Kante an. Sie können voraussetzen, dass keine parallelen Kanten vorkommen und dass alle Gewichte ganze Zahlen mit Absolutbetrag weniger als  $2^{32}$  sind.

**Ausgabeformat:** Das Programm soll eine entsprechende Meldung ausgeben, wenn der Graph unzusammenhängend ist. Ansonsten muss es in der ersten Ausgabezeile das Gewicht des berechneten Baumes ausgeben. Anschließend soll dieser Baum mit `Weighted_Graph::print()` ausgegeben werden.

**Beispiel:** Eine Eingabedatei für einen Graphen mit fünf Knoten und sechs Kanten kann so aussehen:

```
5
0 1 -5
2 4 6
2 1 3
3 2 -2
1 3 0
3 0 6
```

Die Ausgabe des Programms kann dann so aussehen:

```
The following is a minimum spanning tree, with weight -1.
The graph contains vertices 0,...,4 and the following edges:
{0,1} with weight -5
{3,2} with weight -2
{1,3} with weight 0
{2,4} with weight 6
```

**Abgabe:** Vom 7.1.2013 (einem Montag) bis zum 16.1.2013 (einem Mittwoch) im PC-Pool in der Wegelerstraße 6 (Raum E02). Dort müssen Sie sich vorher in ausgehängte Listen mit möglichen Abgabeterminen eintragen. Die Listen hängen ab dem 19.12.2012 aus, und Sie sollten sich dort so bald wie möglich eintragen.

Öffnungszeiten des Help Desks: montags, 12 – 14 Uhr und freitags, 12 – 14 Uhr in Raum N1.002 und donnerstags, 18 – 20 Uhr in Raum N0.003.