Übungen zu WS 95/96

Algorithmen und Programmierung 3

- Blatt 12 -

H. Schweppe / M. Walter

Abgabe: Mo. 5.2.96 bis 12.00

Aufgabe 1 (12 P)

In einem gegebenen Graphen sind die meisten Knoten schwarz gefärbt, wenige sind rot. Gesucht ist die kürzeste Entfernung (= Pfadlänge) eines vorgegebenen Startknotens zum nächstliegenden roten Knoten. Am besten verwendet man dazu die Breitensuche.

Zu entwickeln ist ein Miranda-Programm, das folgende Vorgaben verwendet:

```
graph == [[num]]    || Knoten sind durchnumeriert, Graph ist
                    || als Adjazenzliste repräsentiert

g :: graph
G = .....           || irgendein Beispielgraph
red = [...]         || rote Knoten
distance :: num    ->       num
           || Wurzel        Entfernung
```

`distance` ist die gesuchte Funktion. Es wird am besten mit

```
distance root = dist [root]
```

auf die Funktion `dist` abgestützt: `dist l` ist die Länge des kürzesten aller Wege von einem Knoten aus `l` zu einem roten Knoten (`dist` läßt sich in wenigen Zeilen formulieren).

Aufgabe 2 (10 P)

Gegeben ein ungerichteter Graph G = (V, E). Entwickeln Sie einen O(V+E) Algorithmus, der einen Pfad in G konstruiert, der jede Kante genau einmal in jede Richtung durchläuft. (Trockenübung, Modula2 / Miranda-Implementierung nicht gefordert).

`Letzter Übungszettel mit Punktwertung.`