9. Übung **Informatik B** **SS 08**

Klaus Kriegel **Abgabe: 24.06.2008, 12:00 Uhr**

Aufgabe 1: BFS–Implementierung (3 + 3 + 4 Punkte)

Implementieren Sie eine Unterklasse der Graphklasse aus der 7.Übung. Für alle, die damit Schwierigkeiten hatten, stellen wir eine Musterlösung zur Verfügung. Darin sind auch einige kleinere, aber zur Bearbeitung dieser Aufgabenstellung nützliche Ergänzungen und Modifikationen enthalten.

a) Die Unterklasse heißt `GitterGraph` und hat zwei Attribute `n` und `m`, die ein schachbrettartiges Gitter beschreiben, bei dem in horizontaler Richtung `n` Punkte nebeneinander und in vertikaler Richtung `m` Punkte übereinander liegen. Der Konstruktor soll aus zwei gegebenen Parametern einen solchen Graphen erzeugen, wobei es Ihnen überlassen ist, wie Sie die $n \cdot m$ Knoten des Graphen nummerieren. Es ist sinnvoll, den ersten Konstruktor aus der Musterlösung zu verwenden, mit dem ein kantenloser Graph mit k Knoten erzeugt wird.

b) Implementieren Sie den BFS–Algorithmus und führen Sie diesen auf einem 5×4–Gittergraphen aus (Startpunkt kann selbst gewählt werden). Da vorher bekannt ist, dass nur $n \cdot m$ Knoten in die BFS–Queue eingetragen werden, reicht ein einfaches Array zur Realisierung.

c) Um Ergebnisse von Graphalgorithmen auch in einer Zeichnung darstellen zu können, hat Bogumil eine Klasse `SVG.java` implementiert, mit der man ohne großen Aufwand Dateien erzeugen kann, die sich dann mit einem Browser als Bild anzeigen lassen. Es gibt auch eine Testklasse `SVGTest.java`, die das an einem Beispiel demonstriert. Alles ist auf der VL-Homepage als Ergänzung zur 9. Übung zu finden. Nutzen Sie die Methoden aus `SVG.java`, um den BFS–Baum aus b) zu zeichnen.

Aufgabe 2: Warteschlangen (4+4 Punkte)

a) Implementieren Sie das Interface `Queue<E>` aus der Vorlesung (siehe Skript) mit zyklischen und dynamischen Arrays nach folgenden Regeln:

- Die Default-Größe des Arrays für den Default-Konstruktor ist $N = 10$.
- Hat das Array die Größe M und soll in die Warteschlange ein M-tes Element eingetragen werden, muss die Array-Größe verdoppelt werden.
- Hat das Array die Größe $M \geq 20$ und ist weniger als ein Viertel belegt, wird die Array-Größe halbiert.

b) Testen Sie Ihre Warteschlange mit dem folgenden Beispiel und dokumentieren Sie, wie oft das Array neu angelegt werden muss: Wir durchlaufen die natürlichen Zahlen von 1 bis 1000. Die Zahlen werden nacheinander in die Queue eingefügt, aber bevor man die Zahl n einfügt, löscht sie alle Zahlen vom Anfang der Queue, deren erste Ziffer kleiner oder gleich der ersten Ziffer von n ist.