



# Universität Karlsruhe (TH)

## Institut für Innovatives Rechnen und Programmstrukturen (IPD)

Real Life Programming (Praktikum im SS 2005) <http://www.info.uni-karlsruhe.de/>  
Dipl.-Inf. Michael Beck [beck@ipd.info.uni-karlsruhe.de](mailto:beck@ipd.info.uni-karlsruhe.de)  
Dipl.-Inform. Rubino Geiß [rubino@ipd.info.uni-karlsruhe.de](mailto:rubino@ipd.info.uni-karlsruhe.de)  
Dipl.-Inform. Sebastian Hack [hack@ipd.info.uni-karlsruhe.de](mailto:hack@ipd.info.uni-karlsruhe.de)

Übungsblatt 2

Ausgabe: 19.04.2005

Besprechung: 10.05.2005

### Aufgabe 1: PBQP-Solver (10 Punkte)

Ziel dieser Aufgabe ist es einen Solver für das mathematische Optimierungsproblem PBQP (Partitioned Boolean Quadratic Problem) zu verbessern. Dazu gehört:

**Einlesen der Daten** Der Einleser benötigt momentan die Größe der einzulesenden Vektoren und Matrizen, um vor dem Einlesen des Vektors (der Matrix) ausreichend Speicher anfordern zu können. Zum Beispiel:

```
...  
4  
12 10 11 13  
3  
8 0 1  
...
```

Nun sollen die Einleser Routinen so abgeändert werden, dass sie in der Lage sind, die Vektoren (Matrizen) ohne die vorangestellte Größe einzulesen (Hinweis: Verwende die Techniken aus dem Speicherverwaltungsvortrag). Das Dateiformat soll sich zunächst nicht ändern.

**Optimierung** Bei großen Eingabedateien benötigt der Solver lange Zeit um das Ergebnis zu berechnen, obwohl die Eingabe-Graphen der Theorie zur Folge wesentlich schneller gelöst werden sollen. Finde den Flaschenhals und beseitige ihn, so dass der Solver den von der Theorie vorausgesagten Aufwand zeigt.

Die zum Lösen der Aufgabe nötigen Dateien werden online zur Verfügung gestellt.