

## Übungsblatt 11

Abgabe bis Mittwoch, den 14. Juli um 16 Uhr

### Aufgabe 1 (10 Punkte)

Schreiben Sie eine Methode *findClosestElement*, die für ein gegebenes Element  $x$  und eine gegebene Menge von Elementen dasjenige Element aus der Menge findet, das  $x$  am nächsten ist. Die Menge der Elemente soll dabei aus einer Datei gelesen werden.

Die Elemente sollen dabei einmal ganze Zahlen sein (integer) und einmal Zeichenketten (strings). Wie Sie das Abstandsmaß zwischen zwei Zahlen bzw. zwei Zeichenketten definieren ist Ihnen überlassen, es sollte aber einigermaßen sinnvoll sein. Das einfachste Abstandsmaß zwischen zwei Zahlen ist der Absolutbetrag der Differenz zwischen den beiden Zahlen. Ein einfaches Abstandsmaß zwischen zwei Zeichenketten ist der Absolutbetrag der Differenz der Längen der Zeichenketten. Für ein komplexeres (trotzdem sinnvolles) Abstandsmaß zwischen Zeichenketten gibt es einen Bonuspunkt. Ein Beispiel dafür wären die Levenshtein-Distanz (für eine einfache Definitionen davon, siehe Wikipedia). Wie Sie die Zahlen bzw. Zeichenketten in einer Datei repräsentieren, ist ebenfalls Ihnen überlassen. Am einfachsten und zweckmäßigsten ist wohl eine Zahl bzw. eine Zeichenkette pro Zeile.

Realisieren Sie das Ganze, dem Beispiel aus der Vorlesung folgend, mit einer abstrakten Oberklasse (mit der Implementierung von *findClosestElement*) und konkreten Unterklassen (eine für Zahlen und eine für Zeichenketten). Eine Vorlage für die Klassendeklaration finden Sie im SVN unter *vorlesungen/vorlesung-11/MetricSpaceElement.h*. Sie können, aber müssen sich nicht daran halten. Ein metrischer Raum ist einfach eine Menge von Elementen auf denen ein Abstandsmaß zwischen je zwei Elementen definiert ist, von daher der Klassenname *MetricSpaceElement*.

Wie immer sollen Sie natürlich für jede Methode einen guten Test schreiben. Schreiben Sie für *findClosestElement* eine kleine Testdatei, und schauen Sie ob Ihre Implementierung tatsächlich das nach Ihrem Ähnlichkeitsmaß ähnlichste Element findet. Dazu müssen sie für jeden der beiden Elementtypen (Zahl oder Zeichenkette) einen separaten Test schreiben.

Ein *Main* Programm ist für diese Aufgabe nicht notwendig.

Committen Sie wie gehabt alles ins SVN, in ein neues Unterverzeichnis *uebungsblatt-11*, stellen Sie sicher, dass alles kompiliert, die Tests ohne Fehler laufen, Lint ohne Fehler durchläuft und auch auf Hudson alles funktioniert. Und die *erfahrungen.txt* nicht vergessen.