

Praxis der Software-Entwicklung 2010

Handyprogrammierung mit Java

Prof. Dr. Gregor Snelting – Dennis Giffhorn – Sebastian Buchwald
Lehrstuhl für Programmierparadigmen
Universität Karlsruhe (TH)

Stand: 25.01.2010

In diesem Praktikum sollen Sie eine Tauchcomputer-Simulation realisieren, bei der Sie einen virtuellen Taucher durch die Tiefen des Meeres steuern und alle notwendigen Tauchcomputer-Daten für einen gesunden Unterwasser-Ausflug berechnen. Selbstverständlich dürfen Sie bestimmen, welche Folgen der Zorn der Meere bei uneinsichtigen Tauchern hat...



1 Eine kurze Bemerkung vorab

Dies ist IHR Projekt. Dieses Dokument ist kein Katalog von Aufgaben, der Punkt für Punkt abgearbeitet werden muss, um nachher den Schein zu bekommen, sondern lediglich eine Reihe von Mindestanforderungen, die wir erwarten. Wie IHR Programm nachher aussieht, müssen SIE selbst entscheiden.

2 Motivation

Die Programmiersprache Java existiert nicht nur in ihrer Standardausführung, der Java Standard Edition, sondern auch in einigen verschiedenen Variationen für spezielle Einsatzgebiete, wie etwa Java Card für Smart Cards oder die Java Micro Edition (JavaME) für mobile, vernetzte Geräte wie z.B. Handys. JavaME bietet dazu verschiedene Konfigurationen, um verschiedene Plattformen möglichst gut auszunutzen. In diesem Praktikum sollen Sie mit JavaME, oder alternativ mit Android, einer anderen Java-Variation für mobile Geräte, einen Tauchcomputer entwickeln, der auf mobilen Geräten einsatzfähig ist. Zur 'trockenen' Verwendung des Tauchcomputers sollen Sie zusätzlich einen Simulator entwickeln, der den Tauchcomputer mit Daten füttert und so einen Tauchgang simuliert. Ein solches System könnte zB. zur Planung von Tauchgängen oder zur Schulung eingesetzt werden.

3 Aufgabenstellung

Der Tauchcomputer-Simulator besteht aus den folgenden schwach gekoppelten Komponenten, die miteinander kommunizieren:

- **Aquarium:** Eine Meersimulation: Sie beinhaltet eine Tauchprofilkurve und den Zustand des Tauchers. Es ist ein interaktives System, bei dem die Tauchkurve durch den Benutzer eingegeben/modifiziert werden kann.
- **Tauchcomputer:** Umfasst die Steuerungslogik: Anzeige und Berechnung der Tauchinformationen; ein interaktives System, das Daten vom Aquarium erhält und verarbeitet.

Der Tauchcomputer soll als JavaME- **oder** Android-Applikation realisiert werden, um auf mobilen Geräten lauffähig zu sein. Der Simulator soll auf der JavaSE-Plattform basieren (mind. Java 5) und unter Linux oder Windows vorführbar sein.

3.1 Aquarium

Im Aquarium wird das Tauchprofil per Maus in ein (Zeit, Tiefe)-Koordinatensystem eingegeben. Das Tauchprofil $p(t)$ muss eine stetige Funktion sein mit $p(0) = 0, p(t_{end}) = 0$. Das Aquarium übermittelt $p(t)$ als Folge von Tiefenwerten zu diskreten Zeitpunkten bei festem Zeitintervall (entsprechend 1 sec realer Tauchzeit) an den Tauchcomputer. In jedem diskreten Zeitschritt muss dabei der Zustand des Tauchers aktualisiert und negative Folgen auf die Gesundheit möglichst präzise dargestellt werden. Bedenken Sie dabei, dass die Berechnung des Gesundheitszustands keine Aufgabe des Tauchcomputers, sondern der Meeressimulation ist!

Überlegen Sie sich eine geeignete Möglichkeit, das Tauchprofil zu zeichnen. Soll es vorab gezeichnet werden (mit Korrekturmöglichkeit)? Oder soll es quasi in Echtzeit (mit Zeitrafferfunktion) gezeichnet werden? Hierbei sollen auch Wiederholungstauchgänge möglich sein, d.h. nach dem Auftauchen bleibt der Taucher eine Weile an der Oberfläche und taucht dann wieder ab.

Man kann – wenn Zeit übrig bleibt – dieses Fenster nach Belieben grafisch aufpeppen, mit Fischen, Wellen an der Oberfläche, Mauscursor als Taucher, Lungenriss bei zu schnellem Aufstieg, zerquetschter Taucher bei mehr als 100m Tiefe ... Zumindest muss das Fenster sinnvoll auf unmögliche Tauchprofile (zB unstetige Sprünge, Zeitreisen rückwärts) reagieren.

3.2 Tauchcomputer

Der Tauchcomputer besteht aus einem GUI und einem funktionalen Kern. Das GUI dient zur Darstellung diverser Anzeigen (Zeit, Tiefe, Nullzeit, Decostop, etc.), hat aber auch Eingabemöglichkeiten entsprechend den Knöpfen am Tauchcomputer. Die Funktionalität

soll sich an einen realen Tauchcomputer anlehnen, z.B. von SUUNTO, dessen Bedienungsanleitung Sie auch erhalten werden.

Der funktionale Kern merkt sich für jedes Kompartiment den aktuellen Partialdruck und berechnet daraus sowie aus der nächsten Tiefen- bzw Druckinformation die neuen Partialdrücke sowie die Nullzeit. Es wird dazu die Nullzeitgleichung sowie die Halbwertszeitgleichung verwendet, am besten in der allgemeinen Form für konstante Auf-/Abstiegsgeschwindigkeit. Bei Überschreiten der Nullzeit soll das Dekompressions-Ceiling (Tiefe, die nicht unterschritten werden darf) berechnet werden. An der Oberfläche (Tiefe $< 1\text{m}$) soll die Entsättigungszeit (auf 5% Abweichung zum Normaldruck) angezeigt werden. Bei Wiederholungstauchgängen (innerhalb der Entsättigungszeit) soll nicht neu initialisiert, sondern mit den vorhandenen Werten weitergerechnet werden.

Der funktionale Kern ist gleichzeitig der Sicherheitskern. Er muss strikt das Geheimnisprinzip beachten und eine minimale Schnittstelle haben (ca. 5 Methoden!). Die 1sec-Taktuhr ist nicht Teil des Sicherheitskerns. Der Code für den Sicherheitskern umfasst maximal einige 100 Zeilen!

Da die Zahl der Kompartimente, ihre Halbwertszeit sowie ihre a/b -Faktoren variabel sein sollen, empfiehlt es sich, die entsprechende Information aus einer Konfigurationsdatei zu lesen, auf Plausibilität zu prüfen und an den Sicherheitskern zu übergeben.

Falls die Zeit reicht, können zusätzliche Funktionen gemäß der SUUNTO-Benutzeranleitung realisiert werden (Logbuch, History-Funktion, Datum etc). Beachten Sie, dass die Zusatzfunktionalität nicht mit dem Sicherheitskern interferieren darf.

3.3 Kommunikation

Die beiden Teilsystem sind strikt voneinander getrennt zu halten und sollen über Bluetooth miteinander kommunizieren. Dazu wird Ihnen eine entsprechende Schnittstelle vorgegeben, die es ermöglicht, Aquarien und Tauchcomputer der verschiedenen Gruppen auszutauschen. Die Einhaltung dieser Schnittstelle ist verbindlich!

4 Vorgaben

Neben der verbindlichen Schnittstelle zwischen Aquarium und Tauchcomputer gibt es noch weitere Vorgaben, die Sie in Ihrem Projekt einhalten sollen.

4.1 Architektur

Beide Teilsysteme sollen eine Model-View-Controller Architektur aufweisen.

4.2 JavaME

JavaME gliedert sich in sogenannte Konfigurationen und Profile.

Konfigurationen definieren Minimalanforderungen bezüglich der angestrebten Zielgeräte und machen dementsprechend Einschränkungen und Modifikationen an der Java Virtual Machine und an dem Sprachumfang selbst. Zur Zeit existieren zwei Konfigurationen für JavaME, die *Connected Limited Device Configuration* (CLDC) und die *Connected Device Configuration*(CDC).

Profile definieren zusätzliche APIs für eine speziellere Kategorie von Geräten. Sie werden mit einer Konfiguration kombiniert und ermöglichen so eine Art Feinabstimmung.

Sie sollen mit der CLDC-Konfiguration und dem *Mobile Information Device Profile* (MIDP) arbeiten. Dies stellt eine weit verbreitete Kombination dar. CLDC und MIDP geben Ihnen vor, welche APIs Sie zur Verfügung haben.

Der Tauchcomputer soll in der mit JavaME mitgelieferten Simulationsumgebung `KToolbar` des Sun Wireless Toolkit laufen. Diese Umgebung kann ein Handy emulieren, in dem Ihr Programm ablaufen soll. Beachten Sie, dass damit die Eingabe auf die Handytastatur beschränkt ist.

4.3 Android

Android existiert mittlerweile in der Version 2.1, sie sollen bei Ihrer Entwicklung daher mindestens Version 2.0 unterstützen. Android-Handys werden üblicherweise über einen Touchscreen bedient. Es existieren zwar auch Geräte mit Tastatur; da aber die neueren Modelle weitestgehend auf Tastaturen verzichten, sollen Ihr Programm komplett über Touchscreen bedient werden können.

Der Tauchcomputer soll in der von Android mitgelieferten Simulationsumgebung laufen, wobei das emulierte Gerät die obigen Mindestanforderungen erfüllen soll.

4.4 Qualitätssicherung

Ein Tauchcomputer ist ein sicherheitskritisches eingebettetes System. Deshalb müssen folgende Maßnahmen zur Qualitätssicherung durchgeführt werden:

- JUnit-Testfälle für den Sicherheitskern
- C1-Überdeckung für den Sicherheitskern
- Rechteck-Tauchprofile als Standard-Testfälle; Vergleich mit Tauchtabellen
- Differentialtests: leicht verschiedene Tauchprofile müssen leicht verschiedene Ergebnisse haben
- definierte Reaktion auf falsche Eingaben
- Laufzeitmessungen ($< 1\text{sec}$ pro Berechnungszyklus), Speicherbedarfsmessungen (keine dynamischen Objekte im Sicherheitskern!)

5 Organisatorisches

Für jede Phase des Praktikums muss ein Phasendokument abgegeben werden. Dieses Dokument ist Grundlage für das Kolloquium am Ende jeder Phase, in dem die Gruppe die Ergebnisse der Phase vorträgt.

Zeitlicher Ablauf

- 12.04. - 16.04. Erstes Gruppentreffen mit Betreuer
- 19.04. - 07.05. Pflichtenheft
- 10.05. - 04.06. Entwurf
- 07.06. - 02.07. Implementierung
- 05.07. - 23.07. Validierung
- 26.07. - 30.07. Interne Abnahme + Abschlussveranstaltung

6 Bewertung

Die Benotung Ihres Systems richtet sich nach folgenden Kriterien:

- Qualität aller abgegebenen Dokumente
- Qualität der Kolloquien
- Qualität der Abschlusspräsentation
- Erfüllung der minimalen Leistungsmerkmale (s.o.)
- *sinnvolle* Erweiterungen über diese Merkmale hinaus
- Robustheit des erstellten Programms

Diese Liste hat keine Reihenfolge, die einer Gewichtung entspricht. Es gibt sicherlich weitere Punkte, die als selbstverständlich gelten und sich bei Nichterfüllen negativ auswirken (zB. würde eine völlig unintuitive Bedienbarkeit des Programms zu Abzügen führen).