

Prüfer: Dr. Knoop
Datum: 30.09.2002
Kurs: 01814 objektorientierte Programmierung Kursversion 04/00 (PL 25110)
Dauer: 30 Minuten
Inhalt: Kurstext komplett, zzgl. der beiden englischsprachigen wissenschaftlichen Artikel „A behavioral notion of subtyping“ und „Flexible alias protection“, welche jeweils in den Leittexten referenziert werden (problemloser Download/Beschaffung im Internet).
Note: 1,3

grober Fragen-/Antwortenüberblick:

- 1) Was heißt Paradigma?
Komplex aus Theorien und methodischen Vorgehensweise zur Realisierung eines Vorhabens, im Falle der Programmierung konkret eine Entwurfs- und Programmierungsphilosophie
- 2) Was ist das besondere am OOP-Paradigma und wo kann man es ein- bzw. umsetzen?
Vorbild ist die reale Welt, d.h. man versucht, logisch in sich abgeschlossene Einheiten zu finden und diese angemessen zu abstrahieren. Dabei steht Wiederverwendung, Vererbung und Kapselung (information hiding) im Vordergrund. Daten, Zustände und Verarbeitungsvorschriften werden miteinander kombiniert.
Man kann es beispielsweise einsetzen im Bereich Simulation und graphische Benutzeroberflächen.
- 3) Wofür Spezialisierung?
Verfeinerung eines bestehenden Typs oder einer bestehenden Klasse für einen bestimmten neuen Zweck.
- 4) Was bedeutet überschreiben, überladen und erweitern von Methoden im Rahmen der Vererbung? Wodurch unterscheiden sich überladenen Methoden?
überschreiben = Neudefinition unter identischer Signatur
überladen = Hinzufügen einer zusätzlichen Signatur für eine bestehende Methode
erweitern = Hinzufügen einer neuen Methode
Überladene Methoden unterscheiden sich durch ihre Signatur
- 5) Was ist eine Signatur?
Name der Methode, Typen und Reihenfolge der Parameter [bzw. ()] und Typ des Rückgabeparameters (bzw. void)
- 6) Was heißt SubTyping?
In typisierten OOP-Sprachen wäre ohne SubTyping ein Großteil der Vorteile von Hierarchiebildung verloren, wenn man SubTyping nicht zuließe.
Hinzufügen weiterer Eigenschaften zu einem bestehenden Typ, also Vererbung bestehender Eigenschaften. Der Subtyp ist Teilmenge des Supertyps und kann an allen Stellen im Programm genutzt werden, an denen sonst der Supertyp eingesetzt werden kann. In Java ist auch Vererbung immer mit SubTyping verbunden, also SubClassing.
- 7) Was gilt bzgl. der Typen (und Signatur) bei überschriebenen Methoden?
In Java gilt Gleichheit der Typen. Auf „der grünen Wiese“ der OOP ist jedoch Kovarianz des Rückgabetyps und Kontravarianz der Methodenparameter denkbar, um einen syntaktisch fehlerfreien Ablauf zu gewährleisten.
- 8) Reicht die Typ- und Syntaxkontrolle aus, um ein korrektes Programm zu erzwingen?
Nein, konformes Verhalten basiert nicht nur auf Typkontrolle, sondern auf semantisch korrektem Verhalten.
- 9) Wie begegnen Liskow und Wing der Herausforderung, ein konformes Programm zu schreiben?
Forderung von Vor- und Nachbedingungen für jede Methode sowie Definition sog. Invarianten/Constraints, d.h. die Einführung von „Wohlverhaltensklauseln“.
- 10) Reicht das so annotierte Java-Programm aus, um dann korrekt zu funktionieren?
Teilweise lassen sich Plausibilitätsprüfungen nun automatisieren (neuer Compiler erforderlich)

derlich), jedoch ist eine vollautomatische Validierung von Vor- und Nachbedingungen gegen den Programmcode nicht möglich, so dass nur die Programmierdisziplin geschärft wird.

- 11) Was versteht man unter Parallelität?
Die Existenz von verschiedenen Ausführungssträngen im aktuellen Programm, d.h. die Existenz von Nebenläufigkeit. Auf einem einzelnen Rechner muss dabei der Scheduler für den quasi-parallelen Ablauf sorgen. Bei verteilten Systemen laufen Prozesse generell parallel zueinander ab.
Die Kommunikation zwischen den verschiedenen Ausführungssträngen kann dabei asynchron oder synchron erfolgen.
- 12) Welcher Form von Kommunikation ist diejenige über einen gemeinsamen Speicherbereich zuzuordnen?
Asynchrone Kommunikation; Kontrolle über sog. Monitore.
- 13) Wie können zwei Prozesse miteinander synchron kommunizieren?
beispielsweise Sockets oder entfernter Prozeduraufruf
- 14) Welche Probleme treten bei der Kommunikation mit entfernten Methoden prinzipiell auf?
Übergabeparameter, die keine Werte, sondern Objekte repräsentieren müssen „beachtet werden“: entweder muss das entfernte Objekt komplett lokal vorgehalten oder alle Übergabeparameter an das Ziel transportiert werden. Dabei ist dann immer noch zu klären, was mit „inneren“ Referenzen zu geschehen hat.
- 15) Welche Variante wird in Java mittels RMI realisiert?
Sub-/Skeleton-Mechanismus, also Kommunikation über RMIRegistry. Objektparameter werden serialisiert transferiert?
- 16) Wo ist die besondere Gefahr bzgl. der Parameter?
Es handelt sich bei lokalem und entferntem Objekt zwar um identische, jedoch verschiedene Objekte. Der implizite Parameter (das „this“-Objekt) ist also nicht identisch und muss in der Repräsentation des Objektes entsprechend vorsichtig behandelt werden. Es kann daher passieren, dass eine vormals lokal fehlerfrei funktionsfähige Klasse über RMI nicht mehr fehlerfrei genutzt werden kann.

Fazit

Die Prüfungsatmosphäre war schön ruhig, leider knallt die Sonne ziemlich extrem auf das Informatikzentrum, so dass es im Prüfungszimmer recht warm wird (aber ein netter Fensterblick ☺).

Gemessen an den bisher verfügbaren Prüfungsprotokollen muss man bei Zusammenstellung der Prüfungsvorbereitung nun sicher aufpassen, denn seit Prof. Dr. Poetzsch-Heffter das Lehrgebiet verlassen hat, wird OOP von Dr. Knoop übernommen, wenngleich der Kurs derzeit leider überhaupt nicht im Lehrangebot enthalten ist. Man muss daher auf ältere Kursunterlagen zurückgreifen, um sich vorzubereiten.

Dr. Knoop stellt recht verständliche Fragen, wobei man jedoch nicht immer erraten kann, in welche Richtung die Frage eigentlich abzielt. Dafür gibt's aber ausreichend Hilfestellung, wenn man sich im Kreis zu drehen beginnt. Der Schwierigkeitsgrad wechselt (subjektiv) immer wieder – nicht die gewohnte „ansteigende Kurve“ – was nicht zuletzt daran liegen dürfte, dass tatsächlich das gesamte Kursspektrum angetestet wird.

Java-Kenntnisse sind praktisch nicht erforderlich (keinerlei Sourcen oder Demozeilen), sondern nur die OOP-Theorien.

Der Fragestil und die Vorgehensweise ist etwas ungewohnt im direkten Vergleich zu anderen Prüfungen, jedoch auf angenehme Weise – leider hat er nicht so viele Prüfungsfächer, so dass man hier noch keinen empirischen Vergleich vornehmen könnte.

Als Benotung fiel eine 1,3 ab, garniert mit nützlichen Tipps, wie man weitere Diplomprüfungen noch etwas besser gestalten und absolvieren könnte.