

# LVA 185.276 Analyse und Verifikation (SS 21)

## Leit- und Kontrollfragen IX

Mi, 26.05.2021

*Stoff: Vorlesungsteil V – Kapitel 15*

*Fixpunkte, Transformationen, Optimalität – Unnötige Anweisungen (und ihre Elimination)*

(Ohne Abgabe, ohne Beurteilung)

### Teil V, Kapitel 15 ‘Unnötige Anweisungen (und ihre Elimination)’

1. Anweisungen können in unterschiedlicher Weise als *unnötig* angesehen werden. Nennen Sie einige Beispiele dafür.
2. Die Bestimmung toter Anweisungen in einem Programm ist ein separables Problem, die Bestimmung geisterhafter Variablen nicht. Was ist damit gemeint?
3. Nennen Sie einige andere Beispiele separabler und nichtseparabler DFA-Probleme.
4. Programmtransformationen sollen üblicherweise semantikerhaltend sein. Eine einfache Forderung? Eine einfach zu definierende Forderung? Begründen Sie Ihre Antwort.
5. An verbessernde Programmtransformationen, sog. (Programm-) Optimierungen werden meist konfliktäre Anforderungen gestellt. Illustrieren Sie das anhand einiger Beispiele.
6. Ein Programm ist bzgl. eines Paares aus Vor- und Nachbedingung entweder korrekt oder nicht. Eine Programmtransformation ist in gleicher Weise entweder korrekt oder nicht. Beides sind absolute Eigenschaften. Richtig oder falsch? Begründen Sie Ihre Antwort.
7. Wann heißt eine Anweisung statisch erreichbar? Wann dynamisch erreichbar?
8. Statische und dynamische Erreichbarkeit von Anweisungen ist entscheidbar. Richtig oder falsch? Begründen Sie Ihre Antwort.
9. Senken, Sackgassen, schwarze Löcher. Was verbirgt sich dahinter?
10. Sackgassen und schwarze Löcher in Programmen können stets eliminiert werden. Ein Programmbeobachter wird keinen Unterschied bemerken. Richtig oder falsch? Begründen Sie Ihre Antwort.
11. Wann heißt eine Anweisung *partiell tot*? Illustrieren Sie Ihre Antwort auch anhand eines Beispiels.
12. Illustrieren Sie das Grundmuster der Beseitigung partiell toter Anweisungen.
13. Was sind sog. *Effekte zweiter Ordnung* bei der Beseitigung partiell toter Anweisungen?
14. Partiiell tote Anweisungen in einem Programm lassen sich korrekt und vollständig beseitigen. Richtig oder falsch? Begründen Sie Ihre Antwort.
15. Spezifiziere die Analyse zur Erkennung toter Variablen in einem Programm im Stil von Kapitel 7.
16. Aus welchen Elementartransformationen kann man sich konzeptuell (und praktisch) die Transformation zur Beseitigung partiell redundanter Anweisungen zusammengesetzt denken?
17. Welche Effekte zweiter Ordnung gibt es bei der Beseitigung partiell redundanter Anweisungen?
18. Die Relation *besser* definiert auf der Menge der Programme, die durch Beseitigung partiell redundanter Anweisungen aus einem Ausgangsprogramm hervorgehen, eine partielle Ordnung. Richtig oder falsch? Begründen Sie Ihre Antwort.
19. Spezifiziere die Analyse zur Anweisungshhebung (Anweisungsvorziehung) im Stil von Kapitel 7.
20. In einem Basisblock ist der *Anweisungshhebungskandidat*, die möglicherweise aus diesem Basisblock an eine frühere Programmstelle herausziehbare Anweisung, eindeutig bestimmt. Richtig oder falsch? Begründen Sie Ihre Antwort.

## Teil I – V, Verschiedene Kapitel

1. Welche Funktionen auf dem Verband der Wahrheitswerte sind
  - (a) monoton?
  - (b) distributiv?
  - (c) additiv?
  - (d) inflationär?
  - (e) deflationär?
2. Warum sind nichtmonotone Funktionen als lokale DFA-Semantik von Anweisungen nicht interessant?
3. Warum lässt sich die einfache-Konstanten-Analyse nicht genauso einfach auf parallele Programme ausdehnen wie das für unidirektionale Bitvektoranalysen möglich ist?
4. Sei  $f$  eine beliebige Funktion auf einem vollständigen Verband. Dann ist die zu  $f$  reverse Funktion stets wohldefiniert. Warum?
5. Erläutern Sie die konzeptuellen Unterschiede zwischen natürlicher, strukturell operationeller und denotationeller Semantik.
6. Der denotationelle Semantikdefinitionsstil spricht besonders den Entwerfer einer Programmiersprache an. Warum?
7. Welche anderen wichtigen Interessensgruppen für Programmiersprachen gibt es neben den Entwerfern? Welche Semantikdefinitionsstile sind für diese Gruppen besonders interessant und warum?
8. Aus welchen Teilen besteht die Spezifikation einer Datenflussanalyse? Geben Sie auch ein Beispiel an, um Ihre Antwort zu illustrieren.
9. In welcher Weise muss der Pfadbegriff beim Übergang von sequentiellen zu parallelen Programmen angepasst, erweitert werden?
10. Wie lässt sich beweisen, dass das Konstantenproblem nicht entscheidbar ist?
11. Wie lässt sich beweisen, dass das Schnitt-über-alle-Pfade-Problem nicht entscheidbar ist?
12. Warum ist uns an stärksten Nachbedingungen und schwächsten Vorbedingungen gelegen, nicht aber an schwächsten Nachbedingungen und stärksten Vorbedingungen?
13. Warum gehört axiomatische Semantik nicht zur Reihe von natürlicher, strukturell operationeller und denotationeller Semantik hinzu?
14. Ist die aufwandsbewusste axiomatische Semantik aus Kapitel 5 hilfreich für die Verifikation des Zeitverhaltens harter Echtzeitsysteme? Begründen Sie Ihre Antwort.
15. Neben Laufzeitbedarf ist auch Speicherverbrauch eine wichtige nichtfunktionale Eigenschaft von Programmen. Lässt sich nach dem Muster der aufwandsbewussten axiomatischen Semantik aus Kapitel 5 eine speicherverbrauchsbewusste axiomatische Semantik entwickeln und anhand (von Programmen) der Programmiersprache WHILE demonstrieren und untersuchen? Begründen Sie Ihre Antwort.