

LVA 185.276 Analyse und Verifikation (SS 2020)

Selbsteinschätzungstest 8

Mi, 29.04.2020

Stoff: Vorlesungsteil IV, Kapitel 11 und 12

Chaotische Fixpunktiteration, Unnötige Anweisungen

(Ohne Abgabe, ohne Beurteilung)

Teil IV, Kapitel 11 ‘Chaotische Fixpunktiteration’

1. Warum scheint auf den ersten Blick die Bezeichnung *maximale Fixpunktlösung* als unter gewisse Voraussetzungen berechenbares Gegenstück zur Schnitt-über-alle-Pfade-Lösung unglücklich oder gar falsch?
2. Warum ist auf den zweiten Blick mit dieser Bezeichnung aber doch alles in Ordnung?
3. Wo haben in der Vorlesung Lösungen von Gleichungssystemen bereits eine Rolle gespielt?
4. Was sind *Vektoriterationen*, was *asynchrone Vektoriterationen*?
5. Was garantiert das Fixpunkttheorem von Tarski?
6. Was versteht man unter einer *inflationären* Funktion?
7. Wann würde man entsprechend von einer *deflationären* Funktion sprechen?
8. Gib je ein Beispiel einer inflationären und deflationären Funktion auf den ganzen Zahlen an.
9. Was ist eine *Strategie*? Wann heißt eine Strategie *fair*?
10. Wie hängen die Begriffe *Strategie* und *chaotische Iterationsfolge* zusammen?
11. Wann spricht man von *Familien-Monotonie*? Was für Familien sind gemeint?
12. Was ist ein hinreichendes Kriterium für Familien-Monotonie?
13. Was garantiert das chaotische Fixpunkttheorem?
14. Skizziere einige Anwendungen des chaotischen Fixpunkttheorems.
15. Welche Bewandnis hat es (wohl) mit dem Titelbestandteil “A Folk Tale” in der Arbeit von Jean-Louis Lassez, V.L. Nguyen und Elizabeth A. Sonenberg:
 - *Fixed Point Theorems and Semantics: A Folk Tale*
erschienen in der Zeitschrift *Information Processing Letters*, Band 14, Ausgabe 3, 112-116, 1982?

Teil IV, Kapitel 12 ‘Unnötige Anweisungen’

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 – IV, Verschiedene Kapitel

1. Nur MUT! Welche Doppeldeutigkeit (und Wortspiel) ist darin für parallele Datenflussanalyse verborgen?
2. Für die Behandlung totaler Korrektheit haben wir zwei Kalkülvarianten betrachtet, die sich in der Regel für die while-Schleife unterscheiden. In welcher Weise? Worin liegen die Vor- und Nachteile dieser Regeln?
3. In welcher Weise unterscheiden sich der strukturell operationelle und natürliche Semantikefnitionsstil voneinander?
4. Was bezeichnen *interferenzstabil*, *interferenzlabil* für Eigenschaften?
5. Warum wird für reverse Datenflussanalyse der der induzierenden Datenflussanalyse zugrundeliegende Datenflussanalyseverband um ein neues größtes Element, das sog. *Fehlschlagselement*, erweitert?
6. In welcher Beziehung stehen die *RVUP/RMinFP*-Semantik zueinander? In welcher die *RSUP/RMaxFP*-Semantik?
7. Wie lässt sich das Koinzidenztheorem aus Kapitel 7 beweisen?

8. Wie ändert sich durch
 - (a) Verstärken
 - (b) Abschwächen
 einer Formel p deren Charakterisierungsmenge $Ch(p)$?
9. Im Kontext Hoarescher Logik unterscheidet man zwischen logischen Variablen und Programmvariablen. Warum?
10. Illustrieren Sie die ganzheitliche Sicht auf Datenflussanalyse anhand einer geeigneten Abbildung.
11. Wenn der Gültigkeitsbeweis einer Programmeigenschaft induktiv über die Länge von Berechnungsfolgen geführt worden ist, welche Art von Programmsemantik ist wohl zugrundegelegt gewesen?
12. Nennen Sie einige Beispiele nichtvollständiger Verbände.
13. Warum sind nichtvollständige Verbände für Datenflussanalyse nicht interessant?
14. Welche Funktionen auf dem Verband der Wahrheitswerte sind
 - (a) monoton?
 - (b) distributiv?
 - (c) additiv?
 - (d) inflationär?
 - (e) deflationär?
15. Warum sind nichtmonotone Funktionen als lokale DFA-Semantik von Anweisungen nicht interessant?
16. Warum lässt sich die einfache-Konstanten-Analyse nicht genauso einfach auf parallele Programme ausdehnen wie das für unidirektionale Bitvektoranalysen möglich ist?
17. Sei f eine beliebige Funktion auf einem vollständigen Verband. Dann ist die zu f reverse Funktion stets wohldefiniert. Warum?
18. Erläutern Sie die konzeptuellen Unterschiede zwischen natürlicher, strukturell operationeller und denotationeller Semantik.
19. Der denotationelle Semantikdefinitionsstil spricht besonders den Entwerfer einer Programmiersprache an. Warum?
20. Welche anderen wichtigen Interessensgruppen für Programmiersprachen gibt es neben den Entwerfern? Welche Semantikdefinitionsstile sind für diese Gruppen besonders interessant und warum?
21. Aus welchen Teilen besteht die Spezifikation einer Datenflussanalyse? Geben Sie auch ein Beispiel an, um Ihre Antwort zu illustrieren.
22. In welcher Weise muss der Pfadbegriff beim Übergang von sequentiellen zu parallelen Programmen angepasst, erweitert werden?
23. Wie lässt sich beweisen, dass das Konstantenproblem nicht entscheidbar ist?
24. Wie lässt sich beweisen, dass das Schnitt-über-alle-Pfade-Problem nicht entscheidbar ist?
25. Warum ist uns an stärksten Nachbedingungen und schwächsten Vorbedingungen gelegen, nicht aber an schwächsten Nachbedingungen und stärksten Vorbedingungen?