

Diplomprüfung Säule I

Manuel Klimek

gehalten am 02.09.2002

Parallelprogrammierung	Lengauer
Struktur und Implementierung von Programmiersprachen	Lengauer
Objektorientierte Programmierung	Snelting
Software Werkzeuge	Snelting

Note: 1.7

1 Protokoll

Dieses Protokoll ist im Gegensatz zu allem, was ich bisher gefunden habe, in Erzählform geschrieben. Dies liegt daran, dass die Prüfung selbst schon einige Zeit vorbei ist und ich mich somit nicht mehr an Einzelheiten erinnern kann. Dieses Protokoll gibt also keineswegs alle Prüfungsinhalte wieder, sondern eher einen groben Überblick über den Prüfungsablauf. Im Einzelnen wurde ich natürlich viel mehr gefragt... Viel Vergnügen.

1.1 Prelude

Ich stehe also zitternd vor dem Sekretariat meiner Prüfer, als mein Vorgänger den Raum verlässt. Sichtlich angeschlagen. Lengauer bittet mich sofort herein, offensichtlich um einen Austausch von Prüfungsinhalten zu unterbinden. Doch anstatt mich an den Hinrichtungsort zu begleiten, verläßt er das Büro, um sich "noch mal nen Kaffee zu holen". In der Zwischenzeit eröffnet mir Snelting den Grund der Verzögerung. Ein Prüfling hatte offenbar ein logistisches Problem und war deshalb erst am Ende seiner Prüfungszeit erschienen. Das Adrenalin in meinem Blut strömte derweil ununterbrochen weiter. Nun betrat auch Professor Lengauer den Raum, wobei er seine Tasse frisch gebrühten Kaffee gut gelaunt unter meine Nase stellte und mich angrinste: "Neue Brille? Schick."

1.2 Parallelprogrammierung

In dieser lockere Atmosphäre fühlte ich mich sofort wie zu Hause. Meine Anspannung wich, und ich war wohl etwas unvorsichtig geworden, als ich auf Lengauers unschuldige Frage "So, mit was wollen wir denn anfangen?" mit einem zurückhaltend verwirrten "Öh, weiss nicht?" antwortete. Wie sich später herausstellte war dies mein erster kapitaler Fehler. Ich sah noch den Mundwinkel des Professors hämisch zucken, als er mich prüfend von oben bis unten betrachtete: "Also gut. Dann fangen wir doch mit Parallelprogrammierung an". Nichstahnend fieberte

ich der ersten Frage entgegen. Dann der Schock: "Wir haben da doch einen Haufen Programmiersprachen besprochen, fangen sie mal an...". Nun hatte ich mir eigentlich gedacht, dies wäre eine Säule I Prüfung. Hatte mich also in wochenlanger Arbeit auf Beweisführung in parallelen Modellen, parallele Komplexitätstheorie, Kostenmodelle und das Lengauer-Steckenpferd, den Monitor, vorbereitet. Aber Lengauer unterhielt sich mit mir freudestrahlend über Programmiersprachen, wir philosophierten schliesslich über Java und andere Sinnlosigkeiten des Lebens. Bei MPI wollte er vor allem auf "Konstrukte, die nicht nur kommunizieren, sondern auch was rechnen", und meinte damit Reduce, fragte mich dann auch, was man denn für Operatoren beim Reduce verwenden könnte. Als ich nicht gleich wusste, worauf er hinaus wollte, beantwortete er die Frage gleich selbst: "Assoziativität".

1.3 Compilerbau

Zum Thema Compilerbau wollte Lengauer etwas zum Thema Attributierungen hören. Er fing auch gleich mit Holarith-Konstanten an, deren Probleme ich dann ohne Schwierigkeiten aufzeigen konnte. Auf seine Frage, in welcher Sprache dieses Konzept denn implementiert sei, antwortete ich: "Wird schon irgendeine verquere, kranke Sprache sein...", worauf Snelting kurzerhand einen Lachanfall erlitt. Als dieser sich wieder beruhigt hatte, blickte Lengauer mich ernst an und verkündete: "Das war Fortran". Autsch. Der hatte gegessen. Ich hatte Kapitalfehler Nr. II begangen: Beleidige niemals, niemals in einer Lengauer-Prüfung Fortran. Auch wenn der Chef gleich entschuldigend hinzufügte, dass neuere Fortran-Versionen so etwas nun nicht mehr hätten. Lengauer gab mir noch ein paar Chancen, mit geschichtlichem "Totsünden, und wer sie beging"-Wissen zu brillieren, doch wusste ich auch nichts von Sprachen (ausser Fortran), die keinen Runtimestack implementieren. Snelting warf dann ein, dass meine Generation so etwas wohl nicht mehr kenne, worauf ich anmerkte, dass mir Perl oder Python-Fragen lieber wären. Und wieder sah ich das inzwischen vertraute Bild: zu meiner Linken ein grinsender Snelting und zu meiner Rechten ein etwas grimmiger Lengauer.

1.4 Objektorientierte Programmierung

Snelting fing gleich da an, wo er bei meinem Vorgänger aufgehört hatte: Es ging um Parlsberg-Schwartzbach-Inferenz. Er wollte auch gleich die Inferenzregel für Methodenaufrufe, die ich mit ein wenig Bedenkzeit auch einwandfrei auf das Blatt zauberte. Dann kamen wir zu zeitlichen Invarianten und Transitionssystemen, wobei Snelting immer nur ein Stichwort hören wollte und das auch während der Prüfung gleich zum Ausdruck brachte: "Zeit, genau, Zeit wollte ich hören".

1.5 Software Werkzeuge

Nachdem ich auf die Frage nach RCS und Make wie aus der Pistole geschossen mein Wissen an den Mann brachte, wurde ich für kurze Zeit entlassen, bevor mir meine Note bekanntgegeben wurde.

2 Zusammenfassung

Nun gut, Lengauer hat während der Vorlesung mal verlauten lassen, dass er es "wichtig findet, dass ein Informatikstudent Programmiersprachen zeitlich auch richtig einordnen könne". Habe ich wohl nicht ernst genug genommen. Frühe Fortran Versionen finde ich auch nach dieser Prüfung noch krank und verquer :-)

Snelting war nach meinem Gefühl entgegen der landläufigen Meinung ein angenehmer Prüfer, der zwar sein Stichwort hören will, aber einem auch mal ein wenig Zeit zum Nachdenken gibt. Das habe ich bei Lengauer vermisst.