AFB! - SEHSCHLANGE

Verfasser: Jens Köhler
Quelle: Fachschaft Mathematik/Informatik
Erstellungsdatum: 24.08.1999
Fach: Informatik III/IV (Rechnermodelle und Algorithmen)
Vor-/Hauptdiplom: Vordiplom
Studienfach: Informatik
Prüfer: Prof. Joubert
Beisitzer: Ute Vogel
Prüfling: unbekannt
Datum: 05.06.1991
Dauer: ca. 40 Minuten
Note: 1,0
Prüfungsklima: Sehr angenehm und entspannt; Prof. Joubert fragt sehr gerne die Themen, die er selbst am besten beherrscht (Parallelrechner, Rechnerarchitektur), trotzdem ist auch das, was im Rembold („Informatik") steht, nicht unwichtig. Was in den übungen drankam, wird von den Assis geprüft. Ein bißchen Ausschweifen während der Prüfung oder gar Humor ist durchaus erlaubt und fördert das o.a. Klima.
Prüfungsablauf: Prof. Joubert hat wenige Fragen gestellt und mich dafür ausreden lassen, daher beschreibe ich kurz meine Antworten.

- Wie entscheidet man, ob ein Rechner intelligent ist? (definiere Intelligenz, KI; Turing-Test; common-sense-Frage stellen; beurteilen, ob der Rechner wie ein Mensch geantwortet hat)
- Wie kann man einen Von-Neumann-Rechner verbessern (beschleunigen)? (E/A-Transfer durch Kanal, d.h. E/A-Prozessor; Busbreite/Wortlänge vergrößern; RISC-Architektur, d.h. eingeschränkter Befehlssatz und Speicherzugriff; CACHE-Speicher (Funktionsweise?))
- Zweck eines Betriebssystems? (einfache/zuverlässige/effiziente Bedienung eines Rechners)
- Wie synchronisiert man Prozesse (2 Arten) ? (Semaphoren: Wechselseitiger Ausschluß, Kooperation; durch Nachrichten/Monitore : z.B. bei UNIX die Pipes)
- Was ist in den einzelnen Ebenen angesiedelt (jeweils beschreiben) ?
  M_4**-Ebene::** Auftragssteuerung
  M_3**-Ebene::** Programmallokation; Dateiverwaltung
  M_2**-Ebene::** Betriebsmittelverwaltung; Dispatching
  M_1**-Ebene::** Betriebssystemkern, d.h. Semaphoren, Unterbrechungsverwaltung, physikalischer E/A-Transfer
- Dateiverwaltung: Wie sehen gekettete und invertierte Speicherung einer Datei aus? Jeweilige Vor- und Nachteile? (Man nehme das Beispiel Mitarbeiterdateie und improvisiere)

- Wie kann man Parallelrechner klassifizieren? (z.B. nach Flynn: Wieviele Befehlsfolgen können wieviele Datenfolgen verarbeiten? → SISD-, MISD-, SIMD-, MIMD-Rechner)
- ein praktisches Beispiel: Heute wurden bisher 5 Leute geprüft. Angenommen, es wäre noch Axel Buttchereit als zweiter Beisitzer dabei, und alle drei würden jeweils 15 Minuten prüfen. Wie kann man (unter Mißachtung der Prüfungsordnung) den Prüfungsablaub beschleunigen, wenn man
  1. ein Pipelining simuliert (MISD) ? (Man setze Joubert, Axel und Ute in drei benachbarte Räume und schleuse die Prüflinge im Viertelstundentakt durch diese Räume. Dies verkürzt den Prüfungsablauf auf $(5+2) \cdot 15$ Minuten)
  2. Muliprocessing simuliert (MIMD)? (Man setze die Prüfer wie oben. Während Prüfling 1 bei Joubert hineingeht, gehen Prüfling 2 und 3 bei Axel bzw. Ute rein. Nach 15 Minuten setzt man die Prüflinge um usw. Optimales Scheduling vorausgesetzt, sind dann nur noch $5 \cdot 15$ Minuten erforderlich)

Objektorientiertes Programmieren:: Ute Vogel war mit meinen Antworten sehr schnell zufrieden und hat mich daher nicht ausreden lassen

- Was bewirkt eine Constructor-Prozedur (Anlegen einer VMT mit der Länge des Datensatzes und Adressen der virtuellen Methoden)
- Wie wird der Aufruf einer statischen/virtuellen Methode in Maschinencode umgesetzt (direkter Verweis bzw. Verweis auf VMT-Eintrag)
- Wie erfolgt die Codierung bei Parameterübergabe? (Sichern auf Stack)
- Masterfrage: Warum müssen gleichnamige virtuelle Methoden die gleiche Parameterliste aufweisen? (nur der Aufruf der Methode erfolgt virtuell; die Parameterübergabe hingegen statisch)