

# Rechnersysteme SS 2009

## Aufgabenblatt 7

Vorzustellen vom 8.–12. Juni

### Aufgabe 1

Beschreiben Sie, welche unterschiedlichen Registerklassen es beim Registerfensterkonzept gibt. Welchen Vorteil bietet die dabei verwendete Art der Parameterübergabe gegenüber der Methode bei CISC-Prozessoren und wodurch?

### Aufgabe 2

Eine fünfstufige Pipeline habe eine Schaltlatenz von 3 ns pro Schritt. Ohne Pipeline benötigt der Prozessor 30 ns für die gesamte Abarbeitung eines Befehls.

1. Wie lange benötigt der Prozessor für die Abarbeitung eines Befehls, wenn die Pipeline implementiert wird?
2. Wie lange benötigt dieser Prozessor für die Abarbeitung von zehn Befehlen? Vergleichen Sie den Wert mit dem Wert, der sich beim Pipeline-losen Betrieb ergibt. Wie viele Befehle sind minimal nötig, damit die Pipeline-Version schneller ist?

### Aufgabe 3

1. Nennen und erläutern Sie zwei Ihnen aus der Vorlesung bekannte Techniken, Verzögerungen durch Blasen aufgrund von Sprungbefehlen zu minimieren oder gar zu verhindern!
2. Bei einer CPU mit einer Fünf-Stufen-Pipeline wird in jedem Taktzyklus im Normalfall die Holphase eines Befehls ausgeführt und abgeschlossen. Allerdings führt die Ausführung eines bedingten Sprungs dazu, dass in den folgenden drei Zyklen noch keine weitere Befehlsholphase durchgeführt werden kann. Es bildet sich eine Blase. Um wie viel verzögert sich dadurch die mittlere Ausführungszeit eines Befehls unter der Annahme, dass 20 % aller Befehle bedingte Sprünge sind?

### Aufgabe 4

Angenommen, ein Pipeline-nutzender Computer liest bis zu 20 Instruktionen im Voraus. Im Durchschnitt sind vier davon bedingte Sprünge. Die Wahrscheinlichkeit, dass ein Sprung korrekt vorausgesagt wird, beträgt 90 %. Wie hoch ist die Gesamtwahrscheinlichkeit, dass ein Programmverlauf korrekt vorausgelesen wird?

### Aufgabe 5

Welche der in der Vorlesung vorgestellten Anweisungsabhängigkeiten in einer Pipeline führen möglicherweise zu Problemen? Welche Voraussetzungen müssen dafür gegeben sein? Konstruieren Sie für jede Abhängigkeitsart ein Beispiel in 68k-Assembler-Syntax. Wann entscheidet sich, ob es zu einer Datenabhängigkeit kommt?

### Aufgabe 6

1. Implementieren Sie ein 68k-Assembler-Programm, das ein gegebenes Feld von ein Byte großen Zahlen sortiert. Sollten Ihnen Sortieralgorithmen bekannt sein, dürfen Sie Ihre Vorkenntnisse natürlich nutzen. Nutzen Sie in jedem Fall Unterprogrammtechniken bei der Programmierung.
2. (nur für Nutzer eines interaktionsfähigen Simulators wie Easy68k)  
Erweitern Sie das Programm um Nutzerinteraktion: Der Nutzer gibt zu Beginn die Länge des Zahlenfeldes ein, danach die einzelnen Zahlen. Sobald alle Zahlen eingegeben sind, wird das sortierte Zahlenfeld präsentiert.