

Informatik III

— Übungsblatt 2: Address-Berechnung, C —

Aufgabe 3

5 Punkte

Sei A ein Array von 5 Elementen, wobei jedes Element ein Array aus 7 integers ist:

```
int A[5][7];
```

Bitte geben Sie an, wie viele Bytes der Compiler für das Array reservieren muß und an welcher Adresse das Element $A[i][j]$ gespeichert ist. Gehen Sie dabei davon aus, daß `int` 4 Bytes benötigt. Die Anfangs-Adresse des Arrays sei 1000.

Aufgabe 4

7 Punkte

Schreiben Sie ein C-Programm, das etwa in einem Automaten zur Wechselgeldrückgabe verwendet werden kann. Es soll eine ganze Zahl zwischen 0 und 99 eingelesen werden (der Pfennig-Betrag des Wechselgeldes). Dann soll ausgegeben werden, wie viele Münzen zu 1, 2, 5, 10 und 50 Pfennig zurückgegeben werden sollen. Zur Minimierung der Anzahl von Münzen soll ein "Greedy"-Algorithmus eingesetzt werden: Beginnend bei den größten Münzen werden jeweils so viele Münzen wie möglich gewählt. Sei zum Beispiel der eingeebene Betrag 49 Pfennig. Zuerst wählt der Automat möglichst viele Münzen zu 50 Pfennig, so daß die Summer dieser Münzen noch kleiner als der zurückzugebene Betrag ist. Im Beispiel sind das 0 Münzen. Als nächstes werden 4 Münzen zu 10 Pfennig gewählt, da $4 * 10$ noch kleiner ist als 49 (in C gilt `4 == 49/10`). Anschließend sind noch 9 Pfennig zurückzugeben, also wird eine Münze zu 5 Pfennig gewählt. Danach sind noch 4 Pfennig zurückzugeben, dies ergibt 2 Münzen zu 2 Pfennig. Sie können dieses Programm ohne Kontrollstrukturen (`if`, `while`, etc) schreiben, Daher sollte das C-Beispiel auf den letzten Übungsblatt als Rahmen ausreichen. Wenn Sie wünschen, dürfen Sie aber auch ein `if` einsetzen, um Angaben von 0 Münzen zu unterdrücken.

Abgabetermin:

Bitte geben Sie diese Hausaufgabe spätestens am Donnerstag, den 15. November vor der Vorlesung ab. Abgaben in Gruppen bis zu drei Personen sind erlaubt, aber jedes Gruppenmitglied muß ggf. Fragen über die gesamte Lösung beantworten können.