Technische Universität München
Institut für Informatik
Prof. Dr. Ernst W. Mayr

WS 1993/94
Übungsblatt 7
15. Dez. 1993

Einführung in die Informatik III

(Abgabetermin: 22. Dez. 1993, 13:00Uhr)

Aufgabe 1

Schreiben, beschreiben und testen Sie eine MI-Prozedur (entsprechend den in der Vorlesung angegebenen Aufrufkonventionen), die aus einer gegebenen quadratischen Matrix in situ die transponierte Matrix berechnet. Einziger Parameter der Prozedur ist die Adresse des Felddeskriptors.

Aufgabe 2

Schreiben, beschreiben und testen Sie eine MI-Funktion (entsprechend den in der Vorlesung angegebenen Aufrufkonventionen) zur Berechnung von Fibonacci-Zahlen. Um $F(n)$ zu berechnen, wird die Funktion mit dem Parameter n aufgerufen, der gemäß *call by value* übergeben wird. Die Berechnung in der Prozedur soll gemäß der Definition

$$\begin{aligned} F_0 &= 0,\ F_1 = 1\,, \\ F_n &= F_{n-1} + F_{n-2} \qquad \text{für } n \geq 2 \end{aligned}$$

rekursiv erfolgen.

Aufgabe 3

Sei foo eine Variable, die einer Prozedur als Parameter übergeben wird. Wir betrachten die beiden Operationen

1. R0 $:= foo$ — der Wert von foo wird in das Register R0 geschrieben, und
2. $foo :=$ R0 — der Inhalt des Registers R0 wird der Variablen foo als Wert zugewiesen;

die innerhalb der Prozedur ausgeführt werden. Geben Sie die Operandenspezifikation/Befehlsfolgen an, die diesen beiden Operationen bei der MI entsprechen, falls Parameter gemäß call by value bzw. gemäß call by reference übergeben werden und die in der Vorlesung angegebenen Aufrufkonventionen zugrundeliegen.

Aufgabe 4

Aufträge an den EA-Prozessor EAP1 für eines der beiden Plattenlaufwerke der MI können etwa durch die folgende (Pseudo)Pascal-Prozedur beschrieben werden:

procedure funktion(aktlaufwerk, funkt, f);
  begin
  (str[7 . . . 6], str[5], str[4], str[3 . . . 1]):=(aktlaufwerk, 0, 0, funkt);
  {Start, keine Unterbrechung am Ende}
  {EA–Prozessor wird nebenläufig angestoßen;}
  {Warteschleife mit Terminierung so, daß auch die längsten korrekten Nebenläufe }
  {noch beendet werden können. Es ergibt sich durch Rechnung oder Messung der Wert für z3}
  z3:=8000;
  while (z3 > 0) do
    (f, ende, bereit):=(str[14 . . . 11], str[5], str[0]);
    if (bereit = 0) **then**
      f := gerät_nicht_bereit;
      return;
    if (ende = 1) **then return**; {Fehlerzustand schon in f}
    z3:=z3−1;
  end{of while};
  f:=time_out;
end{of procedure};

Übertragen Sie diese Prozedur in MI. Testen Sie Ihr Unterprogramm, indem Sie es mit $funkt = 5$ für "Sektorkopf lesen" aufrufen. Ergänzende Informationen zur PSP-Steuerung können Sie dem MI-Handbuch entnehmen.

Lese-Aufgabe 5

Informieren Sie sich über die Kommandos `sort`, `cat`, `ps`, `ls`, `cp`, `mv`, `rm`, `touch`, `kill` und `pwd` des Unix-Betriebssystems, indem Sie mittels

$$\text{man } \langle\text{Kommando}\rangle$$

die entsprechende Online-Information abrufen.