Universität Duisburg-Essen
Campus Essen

Prof. Dr. A. Klawonn
Dipl.-Math. O. Rheinbach

Wintersemester 2004/05

15.11.2004

5. Übung zur Vorlesung Numerische Mathematik I

Aufgabe 1 (10 Punkte)

Fast alle modernen Rechner verwenden 64-bit Maschinenzahlen der Form

$$1.b_1b_2b_3\ldots b_{52}\times 2^E,$$

mit $b_i \in \{0,1\}, i = 1,\ldots,52$ und $E = -1022,\ldots,1023$ (IEEE754 Typ "double"). Diese haben im Intervall $[1,2)$ einen Abstand von $\varepsilon = 2^{-52} \approx 2.22\times 10^{-16}$.
Die Zahl ε ist in Matlab als `eps` vordefiniert. Finden Sie für Ihren Rechner die kleinste Zahl x der Form

$$x = 1 + i\varepsilon$$

mit $i \in \mathbb{N}$, so dass

$$x * (1/x)$$

nicht exakt `1` ergibt. Fügen Sie einen Ausdruck bei, der belegt, dass für das gefundene x der Ausdruck `x*(1/x)` nicht 1 ergibt.

Aufgabe 2 (10 Punkte)

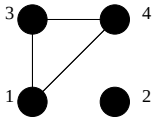
Angenommen, ein archaischer Rechner kann nur das extrem kurze Fließkommaformat

$$\pm b_0.b_1b_2\times 2^E$$

mit $E \in \{-1,0,1\}$ verarbeiten. Dabei wird b_0 explizit (nicht versteckt) abgespeichert. Null wird durch $\pm 0.00\times 2^E$ dargestellt, alle anderen Zahlen sind stets normalisiert. Zeichnen Sie alle in diesem Format darstellbaren Zahlen auf einer Zahlengerade ein. Welches Maschinen-ε hat dieser Rechner? Haben alle Maschinenzahlen den gleichen Abstand?

Aufgabe 3 (10 Punkte)

Ein bekannter griechischer Held entrann dem Labyrinth des Minotaurus nur, weil seine Geliebte ihn mit einem langen Faden ausgestattet hatte, mit dessen Hilfe er den Ausgang wiederfinden konnte.
Ein Labyrinth ist ein Graph mit Räumen V und Gängen E. Den Graphen stellt man am zweckmäßigsten in der Form von Adjazenzlisten dar, die z.B. in einer Matrix abgespeichert werden können. Die Adjazenzlisten zum Graphen



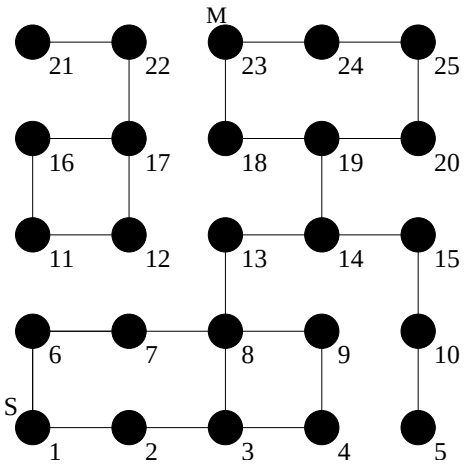
haben dann die Matrixform

$$A = \begin{bmatrix} 3 & 4 & 0 \\ 0 & 0 & 0 \\ 1 & 4 & 0 \\ 1 & 3 & 0 \end{bmatrix}.$$

Dabei bedeutet die erste Zeile, dass vom Knoten 1 jeweils eine Kante zu Knoten 3 und zu Knoten 4 führt. Die darauf folgende Null zeigt an, dass keine weiteren Einträge folgen. Die Nullen in der zweiten Zeile besagen daher, dass von Knoten 2 keine Kanten ausgehen.

Das unten angegebenen Matlab-Programm ariadne.m (→`www.uni-essen.de/numa`) durchsucht das Labyrinth L mit Hilfe der Funktion dfs.m vom Startpunkt S aus. Ergänzen Sie es so, dass der Weg vom Minotaurus (M) zum Startpunkt (S) ausgegeben wird.

L:



ariadne.m

```
global Adj;         % globale Variablen, auf die alle Funktionen zugreifen koennen
global vorgaenger;
global besucht;

% Das Labyrinth L
Adj=[ 2  6  0  0  0; 1  3  0  0  0; 2  4  8  0  0; 3  9  0  0  0; 10 0  0  0  0;
      1  7  0  0  0; 6  8  0  0  0; 3  7  9  13 0; 4  8  0  0  0; 5  15 0  0  0;
      12 16 0  0  0; 11 17 0  0  0; 8  14 0  0  0; 13 15 19 0  0; 10 14 0  0  0;
      11 17 0  0  0; 12 16 0  0  0; 19 23 0  0  0; 14 18 20 0  0; 19 25 0  0  0;
      22 0  0  0  0; 17 21 0  0  0; 18 24 0  0  0; 23 25 0  0  0; 20 24 0  0  0; ];

vorgaenger=zeros(size(Adj,1),1);
besucht=zeros(size(Adj,1),1);
start=1;
dfs(start);
```

dfs.m

```
function dfs(x)
    global Adj;
    global vorgaenger;
    global besucht;

    besucht(x)=1;                % Knoten x als besucht kennzeichnen
    i=1;
    while Adj(x,i)~=0            % es gibt noch Nachbarn
      if besucht(Adj(x,i))==0    % der Nachbar ist unbesucht
        vorgaenger(Adj(x,i))=x;
        dfs(Adj(x,i));           % gehe zum Nachbarn
      else
        % Nachbar schon besucht, brauche nichts zu tun
      end
      i=i+1;
    end
```

Abgabetermin: 22.11.2004 14:00 (im Kasten).