Humboldt-Universität zu Berlin
Institut für Mathematik
Dr. L. Lehmann; Dr. Antoine Laurain, Dipl.Math. Torsten Bosse

Übungsaufgaben zur Vorlesung Mathematikorientierte Computernutzung (SS 11) Serie 4 – erste bewertete Serie

Abgabe bis ?.? 2011

Die Aufgaben werden in die Kategorien **A**: Java-Programmiergrundlagen und Zahlentheorie, **B**: Numerik, Datenstrukturen **C**: HTML, LaTeX, CAS eingeteilt. In jeder Kategorie sind 1/3 der Punkte zu erreichen.

Aufgabe 4.1: Punkte: A8

Schreiben Sie ein Programm, welches nach Eingabe einer geraden Zahl $n \in \mathbb{N}$ einen Diamanten in der Schell ausgibt:

$n = 2:$

```
/\
\/
```

$n = 4:$

```
 /\
/  \
\  /
 \/
```

$n = 6:$

```
  /\
 /  \
/    \
\    /
 \  /
  \/
```

Implementieren Sie dazu in main eine auf Fehler prüfende Eingabe einer geraden Zahl zwischen 2 und 20 und rufen Sie bei erfolgreicher Eingabe eine Ausgabeprozedur Diamant(int n) auf.

Aufgabe 4.2: Punkte: A8, A6

a) Schreiben Sie ein Programm, bei dem der Rechner sich eine natürliche Zahl zwischen 0 und 1000 'denkt' (mit der Funktion `double Math.random()` mit Werten in $[0, 1)$, Skalieren und Runden mit `double Math.floor(double)`) und der Nutzer die Zahl erraten soll. Auf jeden Tipp soll die Antwort „größer" oder „kleiner" gegeben werden. Das Programm soll schließlich die Leistung des Nutzers bewerten, d.h wie oft sich der Abstand zur gesuchten Zahl in etwa halbiert hat und in wievielen Schritten dies nicht passiert ist.

b) Denken Sie sich eine Zahl und lassen Sie den Rechner raten. Versuchen Sie einen Algorithmus zu entwerfen, der so effizient wie möglich ist.

Aufgabe 4.3: Punkte: B4

Berechnen Sie die rekursive Folge $\{x_n\}_{n=0,1,2,\ldots}$, wobei $x_n = x_{n-1}^2$ und $x_0 = 0.5$. Berechnen Sie desweiteren die Folge $y_n = 1 + x_n$ für $n = 0, 1, 2, \ldots$ mit der Folge $\{x_n\}_{n=0,1,2,\ldots}$ von zuvor. Für welches n gilt $x_n = 0$ beziehungsweise $y_n = 1$? Warum gibt es Unterschiede zwischen den n?

Aufgabe 4.4: Punkte: A6

Suchen Sie alle ganzen Zahlen, deren untere Nachbarzahl eine Quadratzahl und deren obere

Nachbarzahl eine Kubikzahl ist. Gestalten Sie ihren Algorithmus möglichst effizient.

Aufgabe 4.5: Punkte: A6

Sei x eine natürliche Zahl. Wenn x gerade ist, wird es durch 2 geteilt. Wenn x ungerade ist, so wird es durch $3x + 1$ ersetzt. Behauptung: Nach endlich vielen Schritten wird $x = 1$. Programmieren Sie das Verfahren rekursiv.
Was kann passieren, wenn die Behauptung nicht wahr ist?

Aufgabe 4.6: Punkte: (Menü: A4) A4, A4, A6, A2, A4, A4

Schreiben Sie ein Rahmenprogramm, welches ein Menü ausgibt (a,b,... oder 1,2,...) und nach Auswahl des Menüpunktes mittels switch-Verzweigung eine der folgenden Aufgaben erledigt:

a) Bestimmen Sie für eine eingegebene Zahl alle Teiler.

b) Bestimmen Sie für eine eingegebene Zahl ihre Zerlegung in Primfaktoren. Hinweis: Alle Primzahlen sind in der Menge $\{2, 3\} \cup \{6k \pm 1 : k = 1, 2, 3, \dots\}$ enthalten.

c) Ermitteln Sie den größten gemeinsamen Teiler (ggT) zweier Zahlen.

d) Bestimmen Sie für zwei eingegebene Zahl das kleinste gemeinsame Vielfache (kgV) (evtl. auch zusammen mit vorigem Punkt).

e) Berechnen Sie die Summe $s(x)$ aller von x verschiedenen Teiler der Zahl x. Überprüfen Sie, ob die Zahl x eine vollkommene Zahl ist (d.h. $s(x) = x$, z.B. $s(6) = 1 + 2 + 3 = 6$). Es gibt wahrscheinlich keine ungeraden vollkommenen Zahlen.

f) Zwei Zahlen x und y heißen befreundet, wenn $s(x) = y$ und $s(y) = x$ gilt. Suchen Sie Paare befreundeter Zahlen (manche sind mit sich selbst befreundet).

Trennen Sie soweit wie möglich Methoden zur Berechnung von Methoden zur Ausgabe der Ergebnisse, so dass z.B. Berechnungsmethoden mehrfach verwendet werden können.

Aufgabe 4.7: Punkte A10

Schreiben Sie eine Klasse `Split`, in deren Namensraum sich die Variablen n und b und Methoden `int getN()` und `int getB()` sowie `void coumpute(int a)` befinden, so dass `coumpute` aus der gegebenen natürlichen Zahl $a = 2^n b$ den Zweierexponent n und den ungeraden Anteil b bestimmt. Verwenden Sie diese Klasse `Split` zur ggT-Berechnung ohne Divisionen (in der Binärdarstellung sind Division und Multiplikation mit 2 einfache Stellenverschiebungen):

$$ggT(2^n x, 2^m y) = 2^{\min(n,m)} \cdot ggT(x, y - x).$$

(Sei etwa $y > x$.) Beachten Sie, dass nach der ersten Reduktion $y - x$ immer eine gerade Zahl ist. Protokollieren Sie alle Zwischenschritte.

Aufgabe 4.8: Punkte A8

Schreiben Sie eine Methode `int ggT(int[] A)`, die den ggT von einem Tupel ganzer Zahlen bestimmt. Verwenden Sie die Beziehung

$$ggT(x_1, \dots, x_n) = ggT(x_1 \bmod x_i, \dots, x_i, \dots, x_n \bmod x_i),$$

falls $x_i \neq 0$ ist. Diese Beziehung kann reihum angewandt werden oder mit dem jeweils kleinsten x_i.
Bestimmen Sie $ggT(201894, 136059, 260015, 372589)$. Kann man mit diesem oder auf ähnliche Weise auch den $kgV(201894, 136059, 260015, 372589)$ finden?

Aufgabe 4.9: Punkte A6; 4 x A4

Schreiben Sie eine Methode `boolean prim(int n)`, die mit möglichst geringem Aufwand testet, ob eine Zahl eine Primzahl ist. Verwenden Sie diese für folgende Teilaufgaben:

a) Euklid: Sei $p_1 = 2,\ p_2 = 3,\ p_3 = 5, \ldots$ die Folge der Primzahlen. Für welche Werte von $n < 100$ ist $\prod_{i=1}^{n} p_i + 1$ eine Primzahl, für welche nicht?

b) Mersenne-Zahlen: Für welche Primzahlen $p < 100$ ist $2^p - 1$ keine Primzahl?

c) Fermat-Zahlen: Suchen Sie die kleinste Zahl der Form $2^{2^n} + 1$, die keine Primzahl ist.

d) Euler: Für welche Zahlen $n < 100$ ist $n^2 + n + 41$ eine Primzahl?

Aufgabe 4.10: Punkte A8

Im Kreis stehen die Kinder 1, 2, ..., n. Nach einem m-silbigen Abzählreim scheidet das jeweils m-te Kind aus, bis niemand mehr im Kreis steht. Ein Programm soll bei vorgegebenen n, m die Reihenfolge des Ausscheidens ausgeben. Beispiel: Bei $n = 6, m = 5$ ergibt sich 5, 4, 6, 2, 3, 1.

Aufgabe 4.11: Punkte A16,A12

a) Auf einem Schachbrett (Felder A1 bis H8) sind 8 Damen so aufzustellen, dass sie sich gegenseitig nicht bedrohen. Die Damen können als globales zweidimensionales Feld `int Damen[8][2]` angelegt werden, eine Prozedur `boolean frei(int k, int x, int y)` prüft, ob das Feld (x, y) von den ersten k Damen nicht bedroht wird. (Im Algorithmus ist immer $k = x - 1$, so dass die Parameterliste reduziert werden kann.)

   Implementieren Sie dazu folgenden Algorithmus:

   1.) Stelle in Reihe A eine Dame auf A1.
   2.) Stelle in der nächsten Reihe eine Dame auf das erste mögliche freie Feld (im ersten Schritt also B3).
   3.) Wiederhole Schritt 2, solange es geht.
   4.) Wenn die letzte Reihe (H) erreicht ist, so hat man eine gültige Stellung gefunden. Die nächste Stellung findet man, indem man die Dame in Reihe H auf das nächste freie Feld setzt.
   5.) Wenn es in einer Reihe kein freies Feld mehr gibt, geht man eine Reihe zurück und sucht dort das nächste freie Feld.
   6.) Das Ende ist erreicht, wenn es für die Dame in Reihe A kein freies Feld mehr gibt.

Wieviele Stellungen gibt es?

b) Probieren Sie mit einen ähnlichen Algorithmus eine maximale Anzahl Springer auf dem Schachbrett zu platzieren.

Aufgabe 4.12: Punkte A16

Programmieren Sie das Spiel 'Mastermind'. Schreiben Sie dazu ein Programm, welches sich eine vierstellige Geheimzahl aus den 4 Ziffern 1,2,3,4 ausdenkt. Der User darf nun 10 Eingabe machen und Versuchen diese Zahl zu erraten. Nach jeder Eingabe soll das Programm angeben, wieviele der Ziffern der Eingabe mit den Ziffern aus der Geheimzahl übereinstimmen und wieviele davon an der richtigen Position sind. Der User hat gewonnen, wenn er innerhalb von 10 Eingaben die richtige Variante gefunden hat.

Aufgabe 4.13: Punkte A8

Benutzen Sie die Turtle Klasse um das Spiel Mastermind zu visualisieren (vgl. Tic tac toe).