Elementare Zahlentheorie mit Maple (01202) WS 2010/11

Klausur am 19.02.2011:

Musterlösungen

Aufgabe 1

Da $\mathrm{ggT}(c,a) = d$ ist, folgt $\mathrm{ggT}(\frac{c}{d}, \frac{a}{d}) = 1$.

Nach Voraussetzung gibt es ein $x \in \mathbb{Z}$ mit $ab = cx$. Es folgt $\frac{a}{d}b = \frac{c}{d}x$, also gilt $\frac{c}{d} \mid \frac{a}{d}b$. Da $\mathrm{ggT}(\frac{c}{d}, \frac{a}{d}) = 1$ ist, folgt, dass $\frac{c}{d}$ ein Teiler von b ist, also $\frac{c}{d}y = b$ für ein $y \in \mathbb{Z}$. Multiplikation mit d ergibt $cy = db$, also $c \mid bd$, die Behauptung.

Aufgabe 2

Sei $n \in \mathbb{N}$ so, dass $n^2 = 11p + 1$ für eine Primzahl p ist. Dann ist $n \neq 1$ und $n \neq 2$, denn $11p+1 > 1^2$ und $11p+1 > 2^2$ für alle Primzahlen p. Es ist also $n > 2$. Ferner gilt $n^2 - 1 = (n-1)(n+1) = 11p$, und da 11 und p Primzahlen sind, besitzt die Primfaktorzerlegung von $n^2 - 1$ genau zwei Primfaktoren. $n^2 - 1$ kann aber auch als Produkt von $n-1$ und $n+1$ geschrieben werden, und beide Faktoren sind größer als 1. Es folgt, dass $n - 1 = 11$ und $n + 1 = p$ ist, oder es ist $n - 1 = p$ und $n + 1 = 11$. Der letzte Fall kann nicht auftreten, denn sonst wäre $p = n - 1 = 9$, ein Widerspruch. Es folgt $n - 1 = 11$ und $p = 13$. Wir haben also gezeigt: Wenn $n^2 = 11p + 1$ für ein $n \in \mathbb{N}$ ist, dann ist $p = 13$. Sei nun umgekehrt $p = 13$. Dann ist $11 \cdot 13 + 1 = 144 = 12^2$. Es folgt, dass $11p + 1$ genau dann eine Quadratzahl ist, wenn $p = 13$ ist.

Aufgabe 3

Es gibt ein $k \in \mathbb{Z}$ mit $a - b = kn$. Sei $d = \mathrm{ggT}(a, n)$. Da d ein Teiler von a und n ist, ist d auch ein Teiler von $b = a - kn$ und n, also von $\mathrm{ggT}(b, n)$. Sei $d' = \mathrm{ggT}(b, n)$. Dann ist analog d' ein Teiler von $a = kn + b$ und n, also von d. Da d und d' beide positiv sind, folgt aus $d \mid d'$ und $d' \mid d$, dass $d = d'$ gilt.

Aufgabe 4

Sei $n = p_1^{e_1} \cdots p_r^{e_r}$ die kanonische Primfaktorzerlegung von n. Dann gilt

$$\varphi(n) = \underbrace{p_1^{e_1-1}(p_1 - 1)}_{<p_1^{e_1}} \cdots \underbrace{p_r^{e_r-1}(p_r - 1)}_{<p_r^{e_r}} < p_1^{e_1} \cdots p_r^{e_r} = n.$$

Aufgabe 5

Eine Folgerung aus dem Kleinen Satz von Fermat ist, dass $x^p \equiv x (\mathrm{mod} p)$ für alle $x \in \mathbb{Z}$ gilt. Sei $a^p + b^p = c^p$. Dann gilt

$$(a^p + b^p) \bmod p = (a^p \bmod p + b^p \bmod p) \bmod p = (a + b) \bmod p.$$

Ferner gilt $c^p \bmod p = c \bmod p$, also $(a + b) \bmod p = c \bmod p$ und damit $a + b - c \equiv 0 (\mathrm{mod} p)$. Es folgt, dass p ein Teiler von $a + b - c$ ist.

Aufgabe 6

Sei $a \in \mathbb{N}$ nicht Summe von zwei Quadraten. Dann gibt es einen Primfaktor p von a, der in der Primfaktorzerlegung von a mit ungeradem Exponenten auftritt. Da $\mathrm{ggT}(a,b) = 1$ ist, tritt dieser Primfaktor in der Primfaktorzerlegung von ab ebenfalls mit ungeradem Exponenten auf. Dies zeigt, dass ab nicht Summe von zwei Quadraten ist.

Aufgabe 7

Da z keine Einheit und keine Gauß'sche Primzahl ist, gibt es $w, w' \in \mathbb{Z}[i]$, die beide keine Einheiten sind, und für die $z = ww'$ gilt. Wir können annehmen, dass $N(w) \leq N(w')$ ist (anderenfalls benennen wir w und w' um). Angenommen, $N(w) > \sqrt{N(z)}$. Dann gilt

$$N(z) = N(ww') = N(w)N(w') \geq (N(w))^2 > (\sqrt{N(z)})^2 = N(z).$$

Dies ist ein Widerspruch, und es folgt $N(w) \leq \sqrt{N(z)}$.

Aufgabe 8

1. Die Eingabe der Prozedur ist $a = 3$, eine natürliche Zahl. In Zeile 3 wird *zähler* auf 0 gesetzt. Der erste Schleifendurchlauf für die for-Schleife, die die Zeilen 4-16 umfasst, ist für $i = 0$. In Zeile 5 wird $i2 := 0^2 \bmod 3 = 0$ und in Zeile 6 *bereitsdrin* auf 0 gesetzt. Da $zähler - 1 = -1$, also kleiner als 0 ist, wird die for-Schleife, die die Zeilen 7-11 umfasst, nicht ausgeführt. Da in Zeile 12 $bereitsdrin = 0$ gilt, wird in den Zeilen 13 und 14 $liste[0]$ auf 0 und *zähler* auf 1 gesetzt. Nun wird die for-Schleife der Zeilen 4-16 für i=1 ausgeführt. In Zeile 5 wird $i2 := 1^2 \bmod 3 = 1$ und in Zeile 5 *bereitsdrin* auf 0 gesetzt. Die for-Schleife der Zeilen 7-11 wird für $j = 0$ ausgeführt. Da die Bedingung $i2 = liste[0]$ nicht erfüllt ist, passiert jedoch nichts. In Zeile 12 ist also *bereitsdrin* immer noch 0, also wird in den Zeilen 13 und 14 zuerst $liste[1]$ auf 1 und dann *zähler* auf 2 gesetzt. Nun wird die for-Schleife aus den Zeilen 4-16 ein letztes Mal für $i = 2$ durchgeführt. In Zeile 5 ist $i2 := 2^2 \bmod 3 = 1$ und in Zeile 6 wird *bereitsdrin* auf 0 gesetzt. Die for-Schleife der Zeilen 7-11 wird nun zunächst für $j = 0$ ausgeführt. Da aber $i2 \neq liste[0]$ gilt, passiert nichts. Anschließend wird die Schleife für $j = 1$ ausgeführt. In Zeile 8 gilt nun $i2 = liste[1]$, also wird in Zeile 9 *bereitsdrin* auf 1 gesetzt. Da die if-Bedingung in Zeile 12 nun nicht erfüllt ist, ist die for-Schleife der Zeilen 4-16 beendet. In Zeile 17 wird nun die Liste $[0, 1]$ ausgegeben.

2. Die Prozedur berechnet für jede natürliche Zahl a eine Liste aller Quadrate modulo a, also aller Werte $i^2 \bmod a$, wobei $0 \leq i \leq a - 1$ gilt.

3. Die Verwendung einer Menge vereinfacht die Prozedur tatsächlich ganz wesentlich:

```
> klausur:=proc(a::posint)
   local i,menge;
    menge:={};
    for i from 0 to a-1 do
      menge:= menge union {i^2 mod a};
    od;
    print(menge);
  end:
```