

Erste Primzahltests

1. Erstellen Sie eine Prozedur *isprime1*, die den folgenden Primzahltest benutzt:

Test 1: n Primzahl \Leftrightarrow Für alle $a \in \mathbb{N}$ mit $1 < a < n$ gilt: $\text{ggT}(a, n) = 1$.

Eingabe: positive Zahl n

Ausgabe: n Primzahl \Rightarrow **TRUE**; n keine Primzahl \Rightarrow **FALSE**

Befehle: ***igcd(n, m)*** $\text{ggT}(n, m)$

- im Detail:
1. Ist $n < 2$, dann ist n keine Primzahl.
 2. Wenn ein a ($1 < a < n$) gefunden wird, so dass $\text{ggT}(a, n) > 1$ ist, dann haben a und n einen gemeinsamen Teiler, der größer als 1 ist. Also besitzt n einen Teiler, der größer als 1, aber kleiner als n ist (a läuft nur von 2 bis $n - 1$), die Zahl n ist demnach keine Primzahl.
 3. Ist $n \geq 2$ und ergibt sich für alle a ($1 < a < n$) immer $\text{ggT}(a, n) = 1$, so besitzt n keine weiteren Teiler als 1 und sich selbst. Also ist n eine Primzahl.

- in MuPAD:
1. ***if-then***-Anweisung mit ***return(FALSE)***
 2. ***for***-Schleife mit ***if-then***-Anweisung mit ***return(FALSE)***
 3. ***return(TRUE)***

Bei der Abarbeitung wird die Prozedur durch ***return*** beendet und der bei ***return*** angegebene Wert wird ausgegeben. Wird ein ***return***-Befehl abgearbeitet, wird die Prozedur sofort verlassen. Weitere Anweisungen der Prozedur werden nicht mehr bearbeitet. Es reicht deshalb, nach dem Testen auf 1. und 2. nur noch ***return(TRUE)*** hinzuschreiben.

Die Befehle innerhalb einer ***for***-Schleife werden nur abgearbeitet, wenn sich die Laufvariable innerhalb des angegebenen Bereiches bewegt. So werden die Befehle der ***for***-Schleife nicht abgearbeitet, wenn *isprime1(2)* aufgerufen wird, weil die ***for***-Schleife dann von 2 hochzählend bis $2 - 1 = 1$ laufen müsste.

Frage: Warum lässt man a von 2 bis $n - 1$ laufen? Dürfte man dies überhaupt? Begründen sie.

Antwort: _____

2. Erstellen Sie eine Prozedur *isprime2*, die den folgenden Primzahltest benutzt:

Test 2: n Primzahl \Leftrightarrow Für alle $a \in \mathbb{N}$ mit $1 < a \leq \lfloor \sqrt{n} \rfloor$ gilt: $\text{ggT}(a, n) = 1$.

Befehle: ***trunc(sqrt(n))*** größte ganze Zahl kleiner gleich \sqrt{n}

3. Vergleichen Sie *isprime1* und *isprime2* bei großen Zahlen ($n > 100000$). Was stellen Sie fest? Für einen Zeitvergleich können Sie den Befehl ***time()*** nutzen. ***time()*** gibt die gesamte Zeit in Millisekunden zurück, die seit Beginn der MuPAD-Sitzung vergangen ist.
 z.B.: **Startzeit := time() ; isprime1(100003); (time() - Startzeit) * ms;**

Ergebnisse: _____

Zusatz 1: Erstellen Sie eine Prozedur ***Primzahlliste2***, die mit Hilfe von *isprime2* eine Primzahlliste erstellt.

Eingabe: Anfangswert und Endwert

Ausgabe: Liste, die alle Primzahlen zwischen Anfangswert und Endwert enthält

Speichern Sie die Arbeit unter ***prime1.mnb*** ab.

Zusatz 2: Informieren Sie sich über Mersennesche Primzahlen. (Linkliste unter www.bernheiden.de)