

Praktikum 3

Ziele

1. Analyse, Implementierung und Anwendung mathematischer Berechnungsformeln
2. Lösungsfindung mittel Brute Force
3. Indirekte Ergebnissuche durch Ausschluss ungeeigneter Werte

Aufgaben

Aufgabe 1

Schreiben Sie ein Programm, welches drei unterschiedliche ganze Zahlen entgegennimmt und die größte bzw. kleinste Zahl bestimmt. Überprüfen Sie die vom Benutzer gemachten Eingaben, ob diese tatsächlich unterschiedlich und ganze Zahlen sind.

Aufgabe 2

Entwickeln Sie ein Programm, welches für eine gegebene positive Zahl n die Fakultät berechnet. Als Fakultät von n bezeichnet man das Produkt aller Zahlen von eins bis n .

Erweitern Sie ihr Programm so, dass es für eingegebene ganzen Zahlen n und k den Binominalkoeffizient berechnet.

Aufgabe 3

Lassen Sie den Benutzer ihres Programms eine Zahl $a \in \mathbb{N}^+$ eingeben. Berechnen Sie im Heron-Verfahren mit $x_0 = \frac{a+1}{2}$ und $x_{n+1} = \frac{1}{2} \cdot \left(x_n + \frac{a}{x_n} \right)$ für $n = 15$ eine hinreichend gute Näherung für \sqrt{a} .

Aufgabe 4

Berechnen Sie alle Primzahlen bis zu einer eingegebenen Obergrenze und geben sie diese auf dem Bildschirm aus (Stichwort: Sieb des Eratosthenes).

Aufgabe 5

Recherchieren Sie den Begriff Brute Force. Notieren Sie zunächst eine Definition von Brute Force mit ihren eigenen Worten im Kopfteil ihres Programms (2-3 Sätze als Kommentar).

Ermitteln Sie alle Kaprekar-Zahlen größer eins bis maximal 800.000 mittels Brute Force. Die Stellen einer quadrierten Kaprekar-Zahl lassen sich so in zwei positive ganze Zahlen zerlegen, dass deren Summe wieder die ursprüngliche Kaprekar-Zahl ergibt.

Beispiele:

$9^2 = 81$. Zerlegt in 8 und 1 ergibt in der Summe wieder 9.

$297^2 = 88209$. Zerlegt in 88 und 209 ergibt in der Summe wieder 297.

Tipp: Machen Sie für die Implementierung mit der Funktionsweise des Perl-Kommandos `length` vertraut.

Aufgabe 6

Bestimmen Sie mittels Brute Force die Anzahl aller rechtwinkligen Dreiecke mit ganzzahligen Seitenlängen kleiner als 500.