Hochschule Bremerhaven
University of Applied Sciences

Fakultät II – Management und Informationssysteme
Informatik
Modul Theoretische Informatik
Prof. Dr.-Ing Henrik Lipskoch

Protokoll zu Aufgabenblatt 11: Team: ti2023_22

Von

Ekane Njoh Junior Lesage Matrikelnmr: 40128
Aguiwo II Steve Matrikelnmer: 40088

Inhalt

I. Aufgabe 1

Bei dieser Aufgabe geht es darum, dass wir uns eine Folge aus der Sammlung OEIS aussuchen, beschreiben und ein Programm in der Sprach LOOP schreiben, dass die Folgenglieder berechnet.

a. OEIS-Auswahl

Es stehen uns für die Aufgabe eine Vielzahl von OIES zur Verfügung, allerdings dürfen wir nur eins davon verwHALTen. Hierbei haben wir uns für OEIS A006037 entschieden.

b. Textuelle Beschreibung und Entsprechung in der realen Welt

Die Folge OEIS A006037 ist eine mathematische Zahlenfolge, die unter dem Namen Weird Numbers bekannt ist. Sie repräsentiert Zahlen, die abudant sind (die Summe ihrer echten Teiler ist größer als die Zahl selbst), aber nicht semiperfekt (keine Teilmenge ihrer Teiler summmiert sich zur Zahl selbst). Ein gutes Beispiel hierfür ist die Zahl 70 : ihrer Teiler sind 1, 2, 5, 10, 14, 35, deren Summe 74 beträgt, aber keine Kombination dieser Teiler ergibt genau 70, In der realen Welt könnten sie al seine Art mathematische Kuriosität betrachtet werden, die zeigt, wie bestimmte Zahlenmuster oder Eigenschaften überraschHALTe und unerwartete Formen annehmen können. Die werden auch Gegenstand theoretischer Untersuchungen in der Zahlentheorie als von praktischer AnwHALTun. Ihre Existenz und Eigenschaft können allerdings zur Untersuchung von Zahlenstrukturen und zur entwicklung von Algorithmen in der theoretischen Informatik beitragen, insbesondere in Bereichen, die sich mit der Natur und Eingenschaften von Zahlen beschäftigen.

Sidney Kravitz hat das gezeigt für k positiv ganze Zahl, Q. eine Primzahl von mehr als 2^k, und

$$R = \frac{2^k Q - (Q+1)}{(Q+1) - 2^k}$$

auch primär und größer als 2^k, dann

$$n = 2^{k-1} Q R$$

ist eine seltsame Zahl.[6] Mit dieser Formel fand er die große seltsame Zahl

$$n = 2^{56} \cdot (2^{61} - 1) \cdot 153722867280912929 \approx 2 \cdot 10^{52}.$$

c. LOOP-Programm

Ein Programm in LOOP zu schreiben, das das n-tes Element unsere Folge berechnet, war eine zu aufwändige Aufgabe aufgrund der Einfachheit der Sprache LOOP, daher haben wir uns min Absprache mit unserem Dozenten dafür entschieden, ein Programm zu schreiben, dass nur die Summe der Teiler einer Zahl berechnet.

```
x_10 := x_10 + 1
// Zähler für mögliche Teiler
x_11 := x_11 + 0
// Summe der Teiler
LOOP x_1 DO
 x_12 := x_167 + 1
 // Aktueller Teilerkandidat
 x_13 := x_133 + 0
 // Hilfsvariable für die Subtraktion
 // Subtrahiere x_12 von x_1, um zu prüfen, ob es ein Teiler ist
 x_14 := x_1 + 0
 // Kopie von x_1 für die Subtraktion
 LOOP x_14 DO
  LOOP x_12 DO
   x_14 := x_14 − 1
  x_13 := x_13 + 1
 // Überprüfe, ob die Subtraktion genau aufging
 x_15 := x_13 + 0
 // Kopie von x_13 für die Multiplikation
 x_16 := x_166 + 0

 // Hilfsvariable für die Multiplikation
 LOOP x_15 DO
  LOOP x_12 DO
   x_16 := x_16 + 1
  // Wenn x_16 gleich x_1 ist, ist x_12 ein Teiler
 x_17 := x_177 + 0
 LOOP x_1 DO
  x_17 := x_17 − 1
 LOOP x_16 DO
  x_17 := x_17 + 1
 // Addiere x_12 zur Summe, wenn es ein Teiler ist
 LOOP x_17 DO
  LOOP x_12 DO
   x_11 := x_11 + 1
 x_10 := x_10 + 1
x_0 := x_11 + 0
// x_11 enthält jetzt die Summe der Teiler von x_1
```

II. Aufgabe 2

Bei dieser Aufgabe sollen wir uns damit beschäftigen die Quersumme einer unsere Matrikelnummer, vorher manuell zu berechnen und im Anschluss daran ein Programm in der Sprache LOOP zu schreiben, dass ebenfalls jede Folgengliednummer der Fole (A006037) aus der vorigen Aufgabe berechnet, mit dem Zusatz, dass für jede Folgengleidnummer n, die durch die berechnete Quersumme teilbar ist, das WHILE-Programm in eine HALTlosscleife geht (also nicht stoppt).

a. Berechnung der Quersumme

Das von uns gewählte Matrikelnummer ist 40128. Wir erhalten also :
$4 + 0 + 1 + 2 + 8 = 15$

Unsere Quersumme ist also 15

b. WHILE-Programm

```
// Eingabe: x_1 (die Zahl, deren Teiler gefunden werden sollen)
// Zähler für mögliche Teiler
x_10 := x_100 + 1
// Summe der Teiler
x_11 := x_11 + 0
WHILE x_1 DO
  // Kopie von x_1 für die Subtraktion
  x_12 := x_1 + 0
  // Hilfsvariable für die Anzahl der Subtraktionen
  x_13 := x_133 + 0
  WHILE x_12 DO
    WHILE x_10 DO
X       _12 := x_12 + 1
    x_13 := x_13 + 1
  // Überprüfe, ob x_12 auf 0 reduziert wurde   x_14 := x_13 − 1
  WHILE x_14 DO
    x_14 := x_14 – 1
    // Addiere x_10 zur Summe
    WHILE x_10 DO
X       _11 := x_11 + 1
  // Nächster Teilerkandidat
  x_10 := x_10 + 1
// Kopie der Summe für die Teilbarkeitsprüfung
x_15 := x_11 + 0
// Teilbarkeitsprüfungszahl
x_16 := x_15 + 0
```

```
WHILE x_15 DO
  WHILE x_16 DO
x         _15 := x_15 – 1
// Überprüfe, ob x_15 auf 0 reduziert wurde
x_17 := x_15 + 0
WHILE x_17 DO
  // Endlosschleife, wenn die Summe durch 15 teilbar ist
  x_17 := x_17 + 0
// x_11 enthält jetzt die Summe der Teiler von x_1
x_0 := x_11 + 0
```

III. Aufgabe 3

Bei dieser Aufgabe handelt es sich darum unser voriges Programm in ein GOTO-Programm zu übersetzen. Wir wissen nämlich aus der Vorlesung, dass dies tatsächlich möglich ist und wiederum.

```
// Eingabe: x_1 (die Zahl, deren Teiler gefunden werden sollen)
x_10 := x_100 + 1
// Zähler für mögliche Teiler
x_11 := x_11 + 0
// Summe der Teiler
M1: IF x_1 := 0 GOTO M2
x_12 := x_1 + 0
// Kopie von x_1 für die Subtraktion
x_13 := x_133 + 0
// Hilfsvariable für die Anzahl der Subtraktionen
M2: IF x_12 = 0 GOTO M3
     P
     GOTO M2
M3: IF x_10 = 0 GOTO M3;
x_12 := x_12 − 1
x_13 := x_13 + 1
GOTO M3
HALT
x_14 := x_13 − 1;
// Überprüfe, ob x_12 auf 0 reduziert wurde
M4: IF x_14 = 0 GOTO M4;
     x_14 := x_14 − 1;
M5: IF x_10 = 0 GOTO M5;
     x_11 := x_11 + 1;
// Addiere x_10 zur Summe
GOTO M5
   HALT
   GOTO M4
  HALT
GOTO M2
   HALT
```

$x\_10 := x\_10 + 1$
$// Nächster Teilerkandidat$
$GOTO\ M1$
$HALT$
$x\_15 := x\_11 + 0$
$// Kopie\ der\ Summe\ für\ die\ Teilbarkeitsprüfung$
$x\_16 := x\_15 + 0$
$// Teilbarkeitsprüfungszahl$
$M6: IF\ x\_15 = 0\ GOTO\ HALT6$
$M7: IF\ x\_16 = 0\ GOTO\ HALT7;$
$x\_15 := x\_15 - 1;$
$GOTO\ M7$
$HALT7$
$GOTO\ M6$
$HALT$
$x\_17 := x\_15 + 0;$
$// Überprüfe, ob\ x\_15\ auf\ 0\ reduziert\ wurde$
$M8: IF\ x\_17 = 0\ GOTO\ HALT8;$
$x\_17 := x\_17 + 0$
$// HALTlosschleife, wenn\ die\ Summe\ durch\ 15\ teilbar\ ist$
$GOTO\ M8$
$HALT$
$// x\_11\ enthält\ jetzt\ die\ Summe\ der\ Teiler\ von\ x\_1$
$x\_0 := x\_11 + 0$

IV. Literaturverzeichnis

https://oeis.org/search?q=A006037&language=english&go=Search

Letzter Zugriff am 21.01.2024