

# Algorithmen und Programmierung

## Xibonacci (Trainingsaufgabe)

Die Xibonacci-Folge definieren wir ähnlich zu Fibonacci-Folge:

$$xib(n) = \begin{cases} s_0, & n = 1 \\ s_1, & n = 2 \\ \dots & \\ s_k, & n = k \\ \sum_{i=n-k}^{n-1} xib(i), & \text{sonst} \end{cases}$$

wobei  $s_i$  gegebene konstante Werte (Signatur) sind und  $n, k \in \mathcal{N}, n > k$  gilt.

**Aufgabe:** Schreiben Sie die folgende C-Funktion:

```
void xib(
    double signature[],
    unsigned int ssize,
    double xseq[],
    unsigned int xsize
);
```

Diese Funktion soll die ersten `xsize` Elemente der Xibonacci-Folge berechnen und im Feld `xseq[]` speichern. Dabei steht die Signatur im Feld `signature[]` zur Verfügung, deren Länge in `ssize`.

Es sind keine verschachtelten Schleifen und Rekursion zulässig. Es ist nicht auszuschließen, dass `xsize` einen kleineren Wert haben kann als `ssize`.

Schreiben Sie ein Testprogramm `xibonacci` und das zum Erzeugen notwendige `Makefile`. Dem Programm werden `ssize+2` Zahlen als Argumente im folgenden Format übergeben:

- `ssize` als ganze Zahl
- `ssize` viele `double`-Werte
- `xsize` als ganze Zahl

Die Ausgabe soll eine Folge von Zahlen mit zwei Nachkommastellen sein.

**Hinweis:** Nutzen Sie `atof()` statt `atoi()` beim Verarbeiten der Kommandozeilenargumente.

### Beispiel:

Aufruf	Ausgabe
<code>./xibonacci 2 0.0 1.0 7</code>	<code>0.00 1.00 1.00 2.00 3.00 5.00 8.00</code>

Bei falscher Anzahl der Parameter muss die folgende Fehlermeldung ausgegeben werden:

`Not enough data!`

### Hinweise zur Aufgabenstellung

Für die Lösung dieser Aufgabe benötigen Sie folgende Grundkenntnisse:

- Schreiben einer `main()`-Funktion
- Schreiben eines `Makefile`
- Kontrollfluss (`if`) in C
- Funktionen in C
- Schleifen in C
- Felder in C
- Lesen von Kommandozeilenargumenten
- Dynamische Speicherverwaltung (`malloc`)
- Benutzung von `gcc`

### Hinweise zur Abgabe

- Erstellen Sie eine ZIP- bzw. TGZ-Archivdatei, welche die geforderten Dateien enthält.
- Fügen Sie dem Archiv keine weiteren Dateien oder Ordner hinzu.
- Reichen Sie Ihre Lösung unter <https://osg.informatik.tu-chemnitz.de/submit> ein.
- Bis zum Abgabende (Deadline), sofern gegeben, können beliebig neue Lösungen eingereicht werden, die die jeweils älteren Versionen ersetzen.
- Ihr Programm muss auf der Testmaschine übersetzbar sein. Deren Details sind auf dem OpenSubmit-Dashboard verfügbar.
- Ihre Lösung wird automatisch validiert. Sie werden über den Abschluss der Validierung per eMail informiert.