



Betreutes Programmieren 6

Aufgabe 6 ** (*Zweitkleinstes Element in zufälliger Zahlenfolge finden*)

Ziel der heutigen Aufgaben ist es, ein Programm mit Benutzung von Zahlenfolgen zu schreiben. Wie letzte Woche sollen zur Erledigung der Aufgaben wieder mehrere kleine Funktionen eingesetzt werden. Geübt werden das Schreiben von Schleifen und von Funktionen für Zahlenfelder und zur Generierung von Zufallszahlen. Falls Sie Erklärungen zu Zahlenfeldern, Zufallszahlen, Schleifen oder Funktionen benötigen, finden Sie diese in den entsprechenden Vorlesungsfolien.

Schreiben Sie ein C-Programm, das ein Zahlenfeld mit Zufallszahlen zwischen 1000 und 9999 initialisiert und dann das zweitkleinste Element berechnet und ausgibt. Die Implementierung soll gewährleisten, dass man das Programm später leicht für nicht-negative Zufallszahlen zwischen anderen Grenzen (maximaler Wert soll dabei höchstens `UINT_MAX - 1` annehmen dürfen) abändern kann. Zerlegen Sie das Programm in die folgenden Funktionen, die Sie dann am Ende im Hauptprogramm benutzen und dadurch das gesamte System zusammensetzen. Lösungen, die nicht dieser Anleitung folgen, werden nicht abgenommen.

a) Funktion zum Generieren von Zufallszahlen zwischen zwei Grenzen

Implementieren Sie eine Funktion

```
unsigned int my_interval_rand(unsigned int u, unsigned int o),
```

die bei jedem Aufruf eine Zufallszahl zwischen `u` und `o` ausgibt. Benutzen Sie für die Implementierung geeignet die Bibliotheksfunktion `rand`, so dass der Startwert später im Hauptprogramm mit der Bibliotheksfunktion `srand` neu gesetzt werden kann.

b) Funktion zum Initialisieren eines Zahlenfeldes mit Zufallszahlen

Implementieren Sie eine Funktion

```
void array_init(unsigned int w[], int n, unsigned int u, unsigned int o),
```

die die Komponenten von `w` mit Hilfe von `my_interval_rand` mit Zufallszahlen zwischen `u` und `o` initialisiert.

c) Funktion zum Finden des zweitkleinsten Elements in einem Zahlenfeld

Implementieren Sie eine Funktion

```
unsigned int array_second_min(unsigned int w[], int n),
```

die den zweitkleinsten in `w` vorkommenden Wert berechnet und zurückgibt, falls dieser existiert - anderenfalls soll `UINT_MAX` zurückgegeben werden.

Beispiele

- In der Folge 1,5,5,1,6 ist 5 der zweitkleinste Wert.
- In der Folge 1,6,2,1,6 ist 2 der zweitkleinste Wert.
- Die Folge 2,2,2,2,2 hat keinen zweitkleinsten Wert.
- Die Folge 5 hat keinen zweitkleinsten Wert.

d) Hauptprogramm

Schreiben Sie eine `main`-Funktion, in der

- ein Zahlenfeld mit 80 zufälligen ganzen Zahlen zwischen 1000 und 9999 angelegt wird,
- das zweitkleinste Element in diesem Feld berechnet wird und
- dieses dann ausgegeben wird, falls es existiert.

Fügen Sie schließlich den Quellcode der obigen Funktionen zu einem funktionierenden Hauptprogramm zusammen. Verwenden Sie für alle konstanten Zahlenwerte geeignete symbolische Konstanten.