Fakultät IV Elektrotechnik und Informatik
Einführung in die Programmierung WS 2021/2022
Manfred Hauswirth / Damien Foucard / Christoph Seifert / Uwe Kuehn

Programmierblatt 1

Ausgabe: *4.11.2021 12:00*
Abgabe: *16.11.2021 08:00*

Thema: *Git, Pseudocode, Insertionsort*

Abgabemodalitäten

1. Alle abzugebenden Quelltexte müssen ohne Warnungen und Fehler auf Deinem Rechner mit dem Befehl `clang -std=c11 -Wall -g` kompilieren.
2. Die Abgabe für den Quellcode erfolgt ausschließlich über unser Git im entsprechenden Branch. Nur wenn ein Ergebnis im ISIS-Kurs angezeigt wird, ist sichergestellt, dass die Abgabe erfolgt ist. Die Abgabe ist bestanden, wenn Du an Deinem Test einen grünen Haken siehst.
3. Du kannst bis zur Abgabefrist beliebig oft neue Versionen abgeben. Lies Dir die Hinweise der Tests genau durch, denn diese helfen Dir Deine Abgabe zu korrigieren.
Bitte beachte, dass ausschließlich die letzte Abgabe gewertet wird.
4. Die Abgabe erfolgt, sofern nicht anders angegeben, in folgendem Branch: `iprg-b<xx>-a<yy>`, wobei `<xx>` durch die zweistellige Nummer des Aufgabenblattes und `<yy>` durch die entsprechende Nummer der Aufgabe zu ersetzen sind.
5. Gib für jede Aufgabe die Quellcodedatei(en) gemäß der Vorgabe ab. Im ISIS-Kurs werden zum Teil Vorgabedateien bereitgestellt. Nutze diese zur Lösung der Aufgaben.
6. Die Abgabefristen werden vom Server überwacht. Versuche Deine Abgabe so früh wie möglich zu bearbeiten. Du minimierst so auch das Risiko, die Abgabefrist auf Grund von „technischen Schwierigkeiten“ zu versäumen. Eine Programmieraufgabe gilt als bestanden, wenn alle bewerteten Teilaufgaben bestanden sind.

Hinweise zum Repository für das Semester

Die Abgaben werden weiterhin mit Git durchgeführt. Nutze dazu das selbe Repository wie im C-Kurs. Eine Anleitung findest Du in unserem ISIS-Kurs auf dem Aufgabenblatt 1 des C-Kurses.

Aufgabe 1 Implementierung Insertion Sort (bewertet)

Implementiere anhand des Pseudocodes in Listing 1 und der Vorlesungsfolien die Funktion `insertion_sort()` in C. Beachte dabei, dass gemäß Konvention Arrayindizes in Pseudocode bei **1** beginnen, in C jedoch bei **0**. Passe die Indizes in Deiner Implementierung also entsprechend an.

Listing 1: Pseudocode Insertion Sort

```
1 InsertionSort(Array A):
2   for j←2 to length(A) do
3     key←A[j]
4     i←j-1
5     while i>0 and A[i]>key do
6       A[i+1] ← A[i]
7       i ← i-1
8     A[i+1] ← key
```

Die Funktion bekommt als Eingabeparameter ein Integer-Array, das sortiert zurückgegeben werden muss. Die zu sortierenden Zahlen werden aus einer Datei eingelesen. Verwende dazu die in der Datei `arrayio.c` vorgegebene Funktion `read_array_from_file`. Gib am Ende Deines Programms das sortierte Array mit der Funktion `print_array` aus.

Mach Dich zunächst mit der Signatur der vorgegebenen Funktionen vertraut, um diese korrekt aufzurufen:

- `int read_array_from_file(int array[], size_t size, char* filename ↪ )`
  Diese Funktion liest eine Folge von maximal `size` vielen Zahlen aus der Datei mit dem Namen `filename` ein und fügt sie in das Array `array` ein.
- `void print_array(int array[], int len)`
  Diese Funktion gibt die ersten `len` vielen Einträge des Arrays `array` mittels printf aus.

In unserem ISIS-Kurs ist ein Beispiel mit zu sortierenden Zahlen in der Datei `zahlen_insertionsort.txt` hinterlegt. Die Werte in dieser Datei dienen nur als Illustrationen. Teste Dein Programm also mit unterschiedlichen Eingaben. Halte Dich dabei an das Format der Datei `zahlen_insertionsort.txt`.

Das folgende Listing zeigt Dir einen beispielhaften Programmaufruf:

Listing 2: Programmbeispiel

```
1 > clang -std=c11 -Wall intro prog_insertionsort.c arrayio.c \
2         -o intro prog_insertionsort
3 > ./intro prog_insertionsort zahlen_insertionsort.txt
4 Unsortiertes Array: 72 -100 1 10 23 -1 55 9 48 2 -53 5
5 Sortiertes Array: -100 -53 -1 1 2 4 9 10 23 48 55 72
```

Nutze die Vorgaben aus unserem ISIS-Kurs. Füge Deine Lösung als Datei `introprog_insertionsort.c` im Abgabebranch in Dein persönliches Repository ein und übertrage die Lösung an die Abgabeplattform.