

Angewandte Informatik

Übungsaufgaben – 17. Dezember 2015

Arrays mit Zahlen

Wir betrachten das folgende Programm:

```
#include <stdio.h>

void f (int *s0, int *s1)
{
    while (*s0 >= 0)
    {
        int *s = s1;
        while (*s >= 0)
            if (*s0 == *s++)
                printf ("%d_", *s0);
        s0++;
    }
    printf ("\n");
}

int main (void)
{
    int a[] = { 10, 4, 3, 7, 12, 0, 1, -1 };
    int b[] = { 7, 14, 0, 8, 9, 22, 10, -1 };
    f (a, b);
    return 0;
}
```

- (a) Was bewirkt die Funktion `f` und warum? (4 Punkte)
- (b) Von welcher Ordnung (Landau-Symbol) ist die Funktion und warum?
Wir beziehen uns hierbei auf die Anzahl der Vergleiche in Abhängigkeit von der Länge der Eingabedaten `s0` und `s1`. Für die Rechnung dürfen Sie beide Längen mit n gleichsetzen, obwohl sie normalerweise nicht gleich sind. (2 Punkte)
- (c) Was passiert, wenn Sie beim Aufruf der Funktion für einen der Parameter den Wert `NULL` übergeben und warum? (2 Punkte)
- (d) Was passiert, wenn Sie das Hauptprogramm wie folgt abändern ([aufgabe-1d.c](#)) und warum?

```
int main (void)
{
    int a[] = { 10, 4, 3, 7, 12, 0, 1 };
    int b[] = { 7, 14, 0, 8, 9, 22, 10 };
    f (a, b);
    return 0;
}
```

(2 Punkte)

- (e) Beschreiben Sie – in Worten und/oder als C-Quelltext –, wie sich die Funktion `f` effizienter gestalten läßt, wenn man die ihr übergebenen Arrays `s0` und `s1` als sortiert voraussetzt. (5 Punkte)
- (f) Von welcher Ordnung (Landau-Symbol) ist Ihre effizientere Version der Funktion und warum? (2 Punkte)