



TECHNISCHE UNIVERSITÄT
CHEMNITZ



Professur Betriebssysteme

Bonusaufgabe 2

Algorithmen und Programmierung

Matthias Werner

Allgemeine Regeln

- Bonusaufgaben sind fakultativ, außer für Studierende des Bachelors Informatik und Kommunikationswissenschaften (B_CC), hier sind sie Vorleistung.
- Die Lösung soll ausschließlich über OpenSubmit abgegeben werden: <https://osg.informatik.tu-chemnitz.de/submit/>
- Bis zum Abgabende (Deadline) können beliebig neue Lösungen abgegeben werden, die die jeweils älteren Versionen ersetzen
- Eine Lösung wird nur bewertet, wenn der Eingangstest von OpenSubmit positiv ausfällt. Das Ergebnis ist im Status Ihrer Abgabe im OpenSubmit dargestellt; Sie werden außerdem bei einem negativen Ergebnis per E-Mail informiert. Bei negativen Eingangstest können Sie (bis zum Abgabende) jederzeit eine neue Lösung einreichen.
- Ihr Programm muss auf der Testmaschine übersetzbar und lauffähig sein. Die exakten Daten der Testmaschine und des Compilers erfahren Sie über den Link „Test Machines“ im OpenSubmit-Dashboard.
- Ihre Abgabe besteht aus einem ZIP- oder TAR-Archiv, welches ausschließlich ihre Quelldatei enthält. Insbesondere sollte das Archiv kein Unterverzeichnis enthalten.
- Zur Übersetzung werden folgende Optionen genutzt:
`-std=c99 -Wall -Wextra -Wpedantic -Werror`

Problem 1 (4 Punkte)

Sei z_i die i . Ziffer einer Zahl. Der positionsinvertierte Wert einer Zahl sei diejenige n -Stellige Zahl, in der an jeder Position i statt z_i nun z_{n-i+1} steht.

Beispiele:

12345 \rightarrow 54321

101 \rightarrow 101

Erweitern Sie folgendes Programm, so dass es den positionsinvertierten Wert des Aufrufparameters berechnet und ausgibt:

```

/* AuP WS 2016/17, Bonusaufgabe 2 */

int printf(const char*, ...);
int atoi(const char*);

// Deklaration der Umkehrfunktion
int reverse(int);

int main(int carg, const char **varg) {
    // ein Parameter uebergeben?
    if (carg != 2) return -1;
    // gib den berechneten Wert aus
    printf("%d\n",reverse(atoi(varg[1])));
    // Rueckgabewert 0 bedeutet: alles okay
    return 0;
}

```

⋮

Definieren Sie `reverse()` und ggf. weitere Funktionen, so sie diese benötigen. Beachten Sie folgende Bedingungen:

- Sie dürfen in Ihrem eigenen Code **keine** Zeichenarrays (Strings) nutzen.
- Sie dürfen (außer den bereits benutzten `atoi()` und `printf()` **keine** Bibliotheksfunktionsrufe benutzen.
- Sie dürfen **keine** Dateien inkludieren (`#include`)
- Sie dürfen **keine** Variablen definieren (Aufrufparameter zählen hier nicht zu Variablen).
- Sie dürfen **keine** Schleifen benutzen.
- Keine Funktion (außer `main()`) darf **Nebeneffekt** (Seiteneffekte) haben.

Falls Sie eines der genannten Konzepte nicht kennen, ignorieren Sie es (und nutzen nur die Konzepte, die Sie kennen).

Sie können diese Rohfassung des Codes auf der Aufgabenseite herunterladen:

<https://osg.informatik.tu-chemnitz.de/lehre/aup/Templates/reverse.c>. Die Quelldatei, die Sie einreichen, **muss** ebenfalls den Namen `reverse.c` haben.

Hinweise:

- Sie lösen die Aufgabe vermutlich am besten, wenn Sie Rekursion benutzen.
- Die Übersetzungsoptionen behandeln alle Warnungen als Fehler. Stellen Sie sicher, dass ihr Code keine Warnungen generiert.
- Sie können bei dieser Aufgabe bis zu vier Punkte erhalten. Entdeckte Betrugsversuche führen zu 0 Punkten.