

*Suit the action to the word,
the word to the action.*
Hamlet, William Shakespeare



UNIVERSITÄT
DES
SAARLANDES

FR Mathematik
Andreas Buchheit

9. Übung zur Vorlesung Programmierung im Sommersemester 2019

Abgabe: Mittwoch, den 19.06.2019 bis spätestens 12 Uhr.

Aufgabe 9.1. (6 Punkte) String Basics

Im Folgenden sollen String Operationen eingeübt werden.

- (a) (2 Punkte) Schreiben Sie eine Prozedur, die einen ANSI String einliest, speichert und ihn invertiert, das heißt die Buchstaben in umgekehrter Reihenfolge wieder ausgibt.
- (b) (2 Punkte) Schreiben Sie eine Prozedur, die mehrfache Leerzeichen in einem String durch ein einzelnes Leerzeichen ersetzt.
- (c) (2 Punkte) Schreiben Sie eine Funktion,

```
int find_and_replace( char*,  
                    size_t length,  
                    const char* word,  
                    const char* replacement);
```

die ein Wort in einem String sucht und es durch einen anderen String ersetzt. Hierbei ist `length` die Länge des String Arrays. Geben Sie die Anzahl an geschriebenen Zeichen zurück, falls ausreichend Platz im String Array ist und `-1` sonst. Sie dürfen annehmen, dass `word` und `replacement` null-terminiert sind.

Aufgabe 9.2. (8 Punkte) Kommandozeilenargumente

In dieser Aufgabe lernen Sie den Einsatz von Kommandozeilenargumenten, indem Sie ein einfaches Programm `cipher` schreiben, welches den ihm nachfolgenden Text mithilfe der Caesar Chiffre codiert. Hierbei wird das Alphabet um c Buchstaben nach rechts permutiert, für $c = 2$ wird z.B. aus 'a' der Buchstabe 'c', aus 'z' der Buchstabe 'b'. Groß- und Kleinschreibung soll erhalten bleiben.

- (a) (2 Punkte) Implementieren Sie zunächst das Parsen von Kommandozeilenargumenten. Implementieren Sie eine Option `-h` die eine Beschreibung der Funktionsweise des Programms liefert und alle möglichen Optionen auflistet und erklärt. Geben Sie eine Fehlermeldung aus, wenn inkorrekte Kommandozeilenargumente angegeben werden.
- (b) (2 Punkte) Implementieren Sie dann eine Debug Option `-d`, die die Anzahl an Kommandozeilenargumenten, den Inhalt des Vektors `argv` sowie alle non-options ausgibt.
- (c) (2 Punkte) Kodieren Sie, falls keine weitere Option gesetzt wird die non-options mithilfe der Caesar Chiffre und $c = 13$. Lassen Sie alle Zeichen, die keine Buchstaben sind unverändert. Geben Sie die kodierte Nachricht aus.
- (d) (2 Punkte) Implementieren Sie die Option `-c`, die als Argument den Shift c erhält. Geben Sie eine Fehlermeldung aus, wenn es sich bei dem Argument nicht um eine Zahl handelt. Implementieren Sie des Weiteren die Option `-r`, die eine kodierte Nachricht entschlüsselt.

Testen Sie ihren Code!