

## Prozedurale Programmierung main()-Argumente

**WICHTIG: Lesen Sie die Aufgabe zuerst komplett durch**

### Aufgabenstellung:

Schreiben Sie ein Programm, das beliebig viele Parameter von der Kommandozeile einlesen kann. Die Kommandozeilenparameter werden mittels `argc` und `argv` an die `main()`-Funktion übergeben. (Der Programmname steht in `argv[0]`)

a) Wird als Programmname „**upper**“ erkannt, so sollen die nachfolgenden Parameter in Großbuchstaben ausgegeben werden. Dazu können Sie die `toupper()` aus der Bibliothek verwenden.

Beispiel:

```
labor@bslpc11:~> ./upper Hallo Welt  
HALLO WELT
```

b) Wird als Programmname „**lower**“ erkannt, so sollen die nachfolgenden Parameter in Kleinbuchstaben ausgegeben werden. Dazu können Sie die `tolower()` verwenden.

Beispiel:

```
labor@bslpc11:~> ./lower Hallo Welt  
hallo welt
```

c) Wird als Programmname „**generatePass**“ erkannt, so sollen sichere Passwörter generiert werden, indem in einem Satz jeweils nur die Anfangsbuchstaben der Wörter verwendet werden.

Beispiel:

```
labor@bslpc11:~> ./generatePass Es sind 120 Meilen bis Chicago wir  
haben Benzin im Tank
```

Passwort: „**Es1MbCwhBiT**“ (Anfangsbuchstaben)

d) Wird als Programmname „**pali**“ erkannt, so sollen die nachfolgenden Parameter in ein Palindrom verwandelt und ausgegeben werden.

Beispiel:

```
labor@bslpc11:~> pali abc lager  
abccba lagerrega l
```

Es dürfen keine globalen Variablen verwendet werden.

Das Programm muss fehlerfrei und ohne Warnungen mit der Compileroption `-Wall` übersetzbar sein.