

OPERATING SYSTEMS BEISPIEL 1

Aufgabenstellung A

Implementieren Sie eine vereinfachte Variante des Unix-Kommandos **expand**.

SYNOPSIS:

```
myexpand [-t tabstop] [file...]
```

Das Programm **myexpand** soll die als Argumente angegebenen Dateien lesen. Ist keine Datei angegeben, soll von *stdin* gelesen werden. Dabei werden auftretende Tabs durch Leerzeichen ersetzt. Die Ausgabe soll auf *stdout* erfolgen. Der optionale Parameter **tabstop** gibt an, an welchen Positionen die Tabs enden sollen (fehlt dieser, ist der Wert 8 anzunehmen).

Anleitung

Lesen Sie die Dateien zeichenweise ein und überprüfen Sie den ASCII-Code des eingelesenen Zeichens. Handelt es sich um ein Tab (**\t**), berechnen Sie die Position **p** des folgenden Zeichens als nächstes Vielfaches von **tabstop** größer der aktuellen Position plus 1:

$$p = \text{tabstop} * ((x / \text{tabstop}) + 1)$$

wobei **x** die Position des Tabs in der aktuellen Zeile und **/** eine ganzzahlige Division (mit Abschneiden der Nachkommastellen) beschreibt.

Testen

Testen Sie Ihr Programm mit mehreren Eingabedateien. Erstellen Sie z.B. eine Testdatei **t1** mit folgendem Inhalt (**\t** steht für einen Tab-Character):

```
1234567890
123\t90
```

Befehl: **myexpand t1** oder **cat t1 | myexpand**
Ausgabe:

```
1234567890
123      90
```

Befehl: **myexpand -t 6 t1**
Ausgabe:

```
1234567890
123    90
```