

## 1.2.5. Vim & Von Befehlen zum Skript II

### Hausaufgabe (100 Punkte)

#### 1. Recap (20 Punkte)

- a. Was ist der Unterschied zwischen Vi(m) und Nano? (5 Punkte)

- b. Beschreibe die folgenden beiden Ausgaben (5 Punkte)

```
$ echo $?  
0
```

```
$ cat -n dummyfile.sh  
cat: dummyfile.sh: No such file or directory  
$ echo $?  
1
```

- c. Was passiert, wenn du versuchst, eine Variable in Bash mit einem Leerzeichen zuzuweisen, z.B. username = Carol ? (5 Punkte)

#### 2. Navigation in vi (20 Punkte)

- a. Welcher Befehl in vi springt zum Anfang der aktuellen Zeile? (1 Punkt)

- b. Welcher Befehl springt im vi-Editor zum Anfang der Datei? (1 Punkt)

- c. Welcher Befehl springt im vi-Editor zum Ende der Datei? (1 Punkt)

- d. Welcher Befehl sucht im vi-Editor nach einem Textmuster vorwärts in der Datei? (1 Punkt)

- e. Wie scrollt man in vi eine Seite nach unten? (2 Punkte)

- f. Wie beende ich den Insert-Modus und gehe in den Navigationsmodus und dann in den Befehlsmodus in vi? (4 Punkte)

- g. Wie gehe ich in den Insert-Modus in vi? (2 Punkte)

- h. Wie schließe ich eine Datei in vi nachdem ich sie bearbeitet habe, ohne sie speichern zu wollen? (4 Punkte)

- i. Wie schließe ich eine Datei in vi nachdem ich sie bearbeitet habe mit anschließender Speicherung? (4 Punkte)

### 3. Bedingungslogik in Bash-Skripten (30 Punkte)

Schreibe ein Bash-Skript, das überprüft, ob eine Variable mit dem Namen \$num einen Wert größer als 10 hat. Wenn der Wert größer als 10 ist, gib "Nummer ist

größer als 10" aus. Andernfalls gib "Nummer ist nicht größer als 10 aus". Gib bitte einen Screenshot vom Bash-Skript ab.

4. **Bash-Skript mit zwei Argumente (30 Punkte):**

Schreibe ein einfaches Skript, das überprüft, ob genau zwei Argumente übergeben werden. Wenn ja, gib die Argumente in umgekehrter Reihenfolge aus. Gib bitte einen Screenshot vom Bash-Skript ab.

5. **Bonus: For-Schleife in Bash-Skripten (20 Punkte):**

Schreibe ein Bash-Skript, das über alle Argumente iteriert, die an das Skript übergeben wurden. Gib außerdem die Anzahl an Argumenten aus. Gib bitte einen Screenshot vom Bash-Skript ab.