



Aufgabe 1

Starten Sie eine PowerShell als Administrator.

- Sortieren Sie den Verzeichnisinhalt des Windows-Verzeichnisses (nur Dateien) absteigend nach Größe und exportieren Sie das Ergebnis in die CSV-Datei *dateien.csv* im aktuellen Verzeichnis.
- Finden Sie heraus, wie viele Dateien mit der Endung *.exe* im Windows-Verzeichnis liegen. Geben Sie auch an, wie groß diese Dateien insgesamt und durchschnittlich sind.

Aufgabe 2

- Lösen Sie das Knuth-Problem (s. Folien zum Kapitel 3) mithilfe von PowerShell. Verwenden Sie zum Testen die Textdatei *words.txt*.

Hinweise:

- Zum Parsen der Textdatei in einzelne Wörter können Sie den Ausdruck `(Get-Content words.txt) -Split '\W+'` am Anfang der Pipeline nutzen.
 - Den Parameter *n* können Sie als erstes Commandline-Argument über `$Args[0]` ansprechen.
- Vergleichen Sie Ihre Lösung mit der UNIX Shell-Lösung von Doug McIlroy. Welche Besonderheiten und Unterschiede zwischen den Shells finden Sie?

Aufgabe 3 (Bonus)

Schreiben Sie eine Pipeline, die zu einem beliebigen Cmdlet den Typ des Rückgabeobjektes (und nur diesen) ausgibt. Das Beispiel zeigt als Input das Cmdlet *Get-LocalUser*. Das zugehörige Rückgabeobjekt ist vom Typ *Microsoft.PowerShell.Commands.LocalUser*. Der Output ist abgeschnitten.

```
PS C:\Users\anr> Get-LocalUser | Get-Member

TypeName: Microsoft.PowerShell.Commands.LocalUser

Name      MemberType Definition
----      -
Clone     Method     Microsoft.PowerShell.Commands.LocalUser Clone()
Equals    Method     bool Equals(System.Object obj)
GetHashCode Method     int GetHashCode()
```

Ihr Befehl soll genau folgende Ausgabe liefern:

```
Microsoft.PowerShell.Commands.LocalUser
```

Verwenden Sie zum Testen verschiedene Ihnen bekannte Cmdlets und lassen Sie deren Typ ausgeben.