

# Betriebs- und Kommunikationssysteme | Zettel 1

Studenten: Evghenii Orenciuc, Jonathan Rex

Tutor: Abraham Söyler

## Aufgabe 1

### Begriffe

1. Instruction Set: Eine Sammlung von Processorbefehlen
2. Process: ein laufendes Programm
3. File: ein Informationskontainer
4. System Call: eine Verfahren fuer ein Programm mit OS-Kernel zu kommunizieren
5. Multitasking: parallele Ausfuehrung von mehreren Programmen auf einem Rechner/Prozessor
6. Protection Rings: ein hierarchisches Schutzmechanismus von Daten und Funktion eines Betriebssystems

### Schichtenmodell

- a)
1. Nutzer
    - 1.1 Student
    - 1.2 Professorin
  2. Anwendungsprogramme
    - 2.1 Datenbanksystem
    - 2.2 Texteditor
    - 2.3 Graphikprogramm
    - 2.4 Compiler
    - 2.5 Debugger
    - 2.6 Texteditor
  3. Betriebssystem
    - 3.1 Scheduler
    - 3.2 Assembler
    - 3.3 Speichermanagement
    - 3.4 Dateisystem
    - 3.5 BIOS
    - 3.6 Cache
    - 3.7 Gerätemanagement
  4. Hardware
    - 4.1 CPU
    - 4.2 Transistor
    - 4.3 Fetsplatte
- b) Betriebssystem ist eine spezielle Software (und die einzige), die direkt mit Hardware arbeitet.

## Aufgabe 2

```
#include <stdio.h>
#include <stdlib.h>

// no check if file is empty, or doesn't have enough data for calculation
int main(void)
{
    FILE *fp = fopen("test.dat", "r");

    if (fp == NULL)
        return 1;

    int fileSum = 0;
    int calculatedSum = 0;
    int val1 = 0, val2 = 0;

    fscanf(fp, "%d", &fileSum);

    while((fscanf(fp, "%d*d", &val1, &val2)) != EOF) {
        calculatedSum += val1 * val2;
    }

    fclose(fp);

    int result = (fileSum == calculatedSum) ? 0 : 1;
    printf("Result: %d\n", result);

    return result;
}
```

### Results:

```
# test.dat
200
1*100
2*50

gcc -std=c11 -Wall -Wextra -pedantic u1v2.c
./a.out
Result: 0
```

u1

```
# test.dat
400
5*2
100*7
```

```
gcc -std=c11 -Wall -Wextra -pedantic u1v2.c
./a.out
Result: 1
```

```
makefile:7: recipe for target 'run' failed
make: *** [run] Error 1
```