

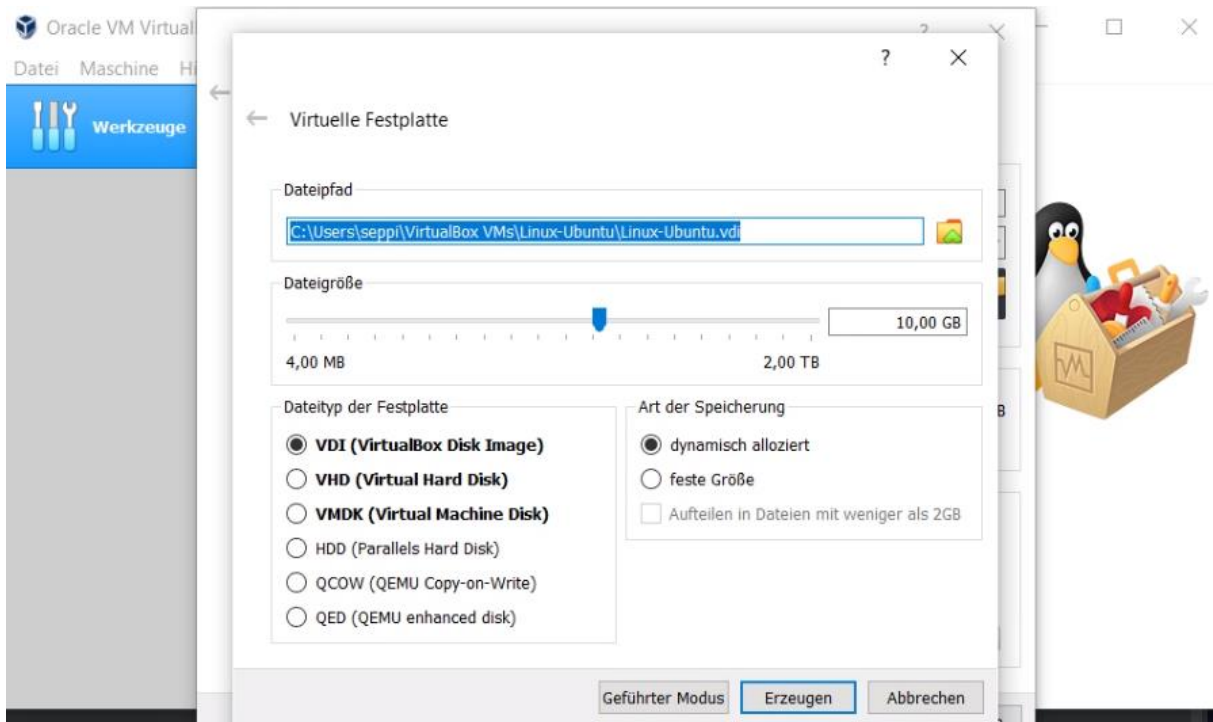
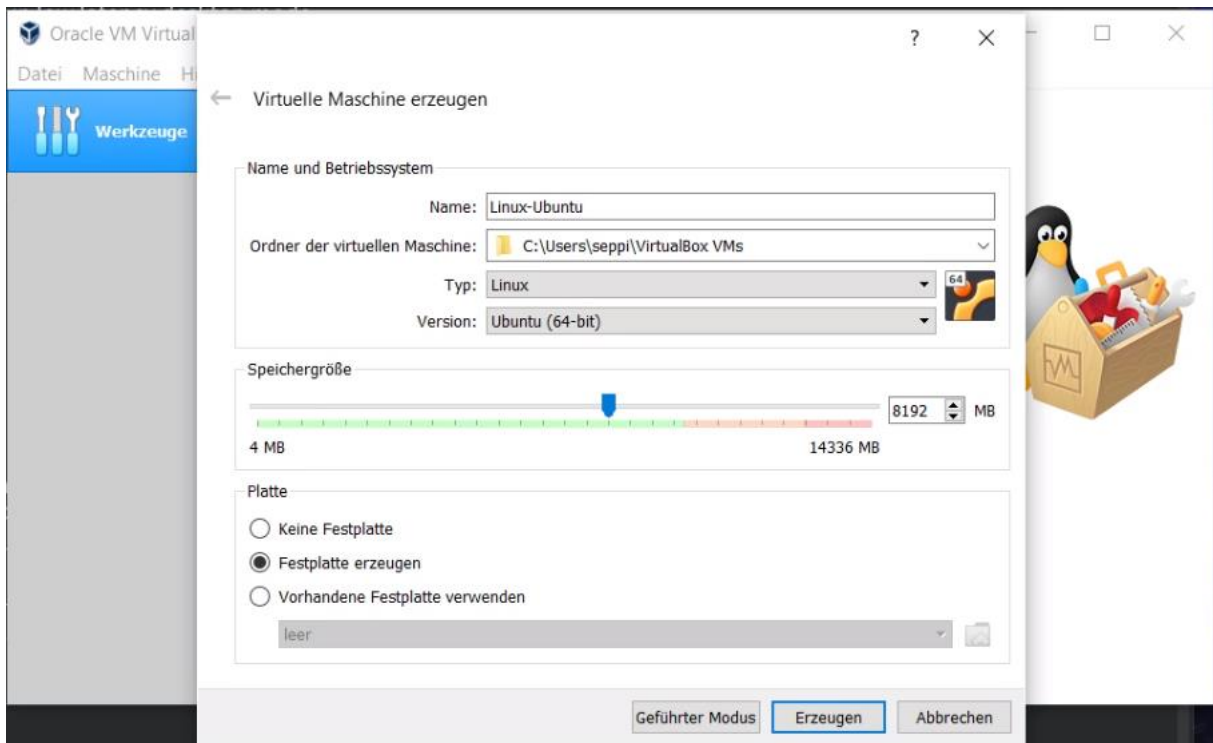
Betriebssysteme Praktikum 1

-

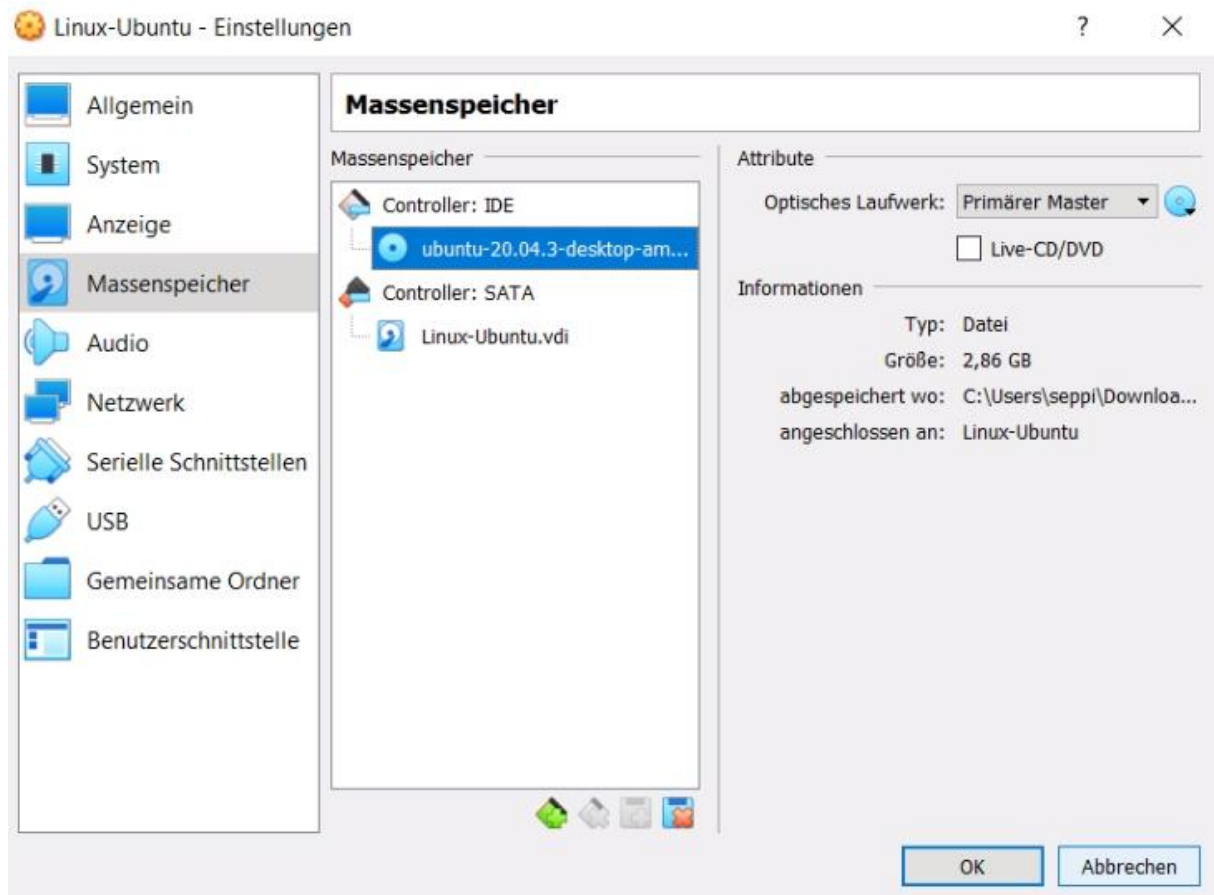
Jörg Quick 762025 Sebastian Zill 769544

Teil 1:

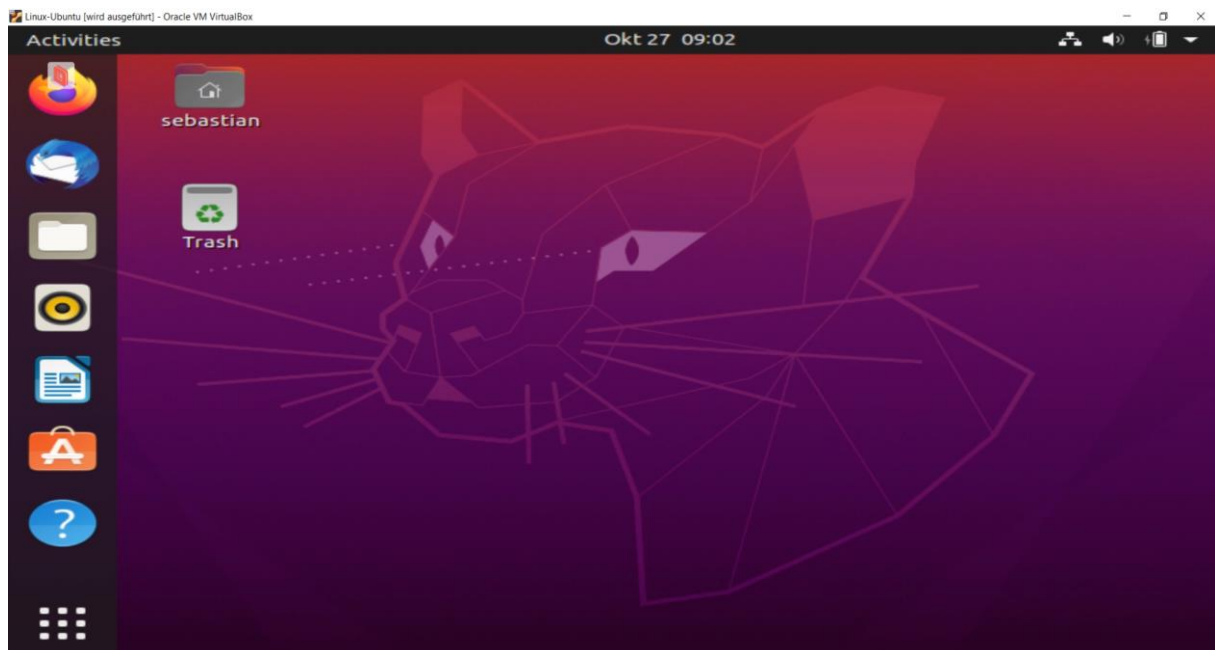
- VM Aufsetzen - Linux Ubuntu über Virtualbox oracle



Bootmedium hinzufügen (ubuntu .iso Datei hinzufügen.)



Ubuntu VM einrichten (First-Time-Installation und Tastatur Layout)



Install gcc Compiler + Debugger

- `sudo apt-get update`
- `sudo apt-get install build-essential gdb g++ (g++ Compiler install)`
- `g++ --version` (Compiler prüfen)

Install IDE (Codeblocks)

- `sudo apt-get install codeblocks`

Teil 2.

Schreiben Sie eine Funktion **funcMem**, die einen großen Array von Integern zuerst allokiert und dann jedes einzelne Arrayelement anspricht (z.B. durch eine Zuweisung).

Schreiben Sie eine rekursive Funktion **funcRec**, die im Wesentlichen sich selbst aufruft.

Code siehe: https://code.fbi.h-da.de/istjoquic/BS_Praktikum_1

1. Dokumentieren Sie die maximal nutzbare Größe an Hauptspeicher sowie die maximal nutzbare Stackgröße, getrennt als experimentell ermittelte bzw. aus dem Betriebssystem über `getrlimits` ausgelesene Größen.

Hauptspeicher: Zugewiesener Speicher an VM - Auslesen mit „`ulimit -a`“:

```
sebastian@sebastian-VirtualBox:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority     (-e) 0
file size               (blocks, -f) unlimited
pending signals         (-i) 31605
max locked memory       (kbytes, -l) 65536
max memory size         (kbytes, -m) unlimited
open files              (-n) 1024
pipe size               (512 bytes, -p) 8
POSIX message queues    (bytes, -q) 819200
real-time priority      (-r) 0
stack size              (kbytes, -s) 8192
cpu time                (seconds, -t) unlimited
max user processes      (-u) 31605
virtual memory          (kbytes, -v) unlimited
file locks              (-x) unlimited
```

	DOKUMENTATION	EXPERIMENT
HAUPTSPEICHER	unlimited	Unlimited (Programm läuft bis zu Segmentation-Fehlerfällen)
STACKGRÖßE	8388KB	3482KB (Segmentation fault)

```
#define ARRLEN 50000
#define INTERVAL 10000
Segmentation fault (core dumped)
Process returned 139 (0x8B)   execution time : 0.248 s
```

2. Dokumentieren Sie die von Ihrem Programm benötigte User bzw. System CPU Time und deren Verhältnis in Abhängigkeit von *n* bzw. *m* (s.o.). Dazu können Sie eine Tabelle oder ein Diagramm verwenden.

ARRLEN = 25000

N = INTERVALL

	N	USER CPU TIME	SYSTEM CPU TIME
	1	105663ms	246718ms
	10	12696ms	20945ms
	100	3825ms	6160ms
	1000	2957ms	2251ms
	10000	3681ms	1750ms
	25000	2050ms	-