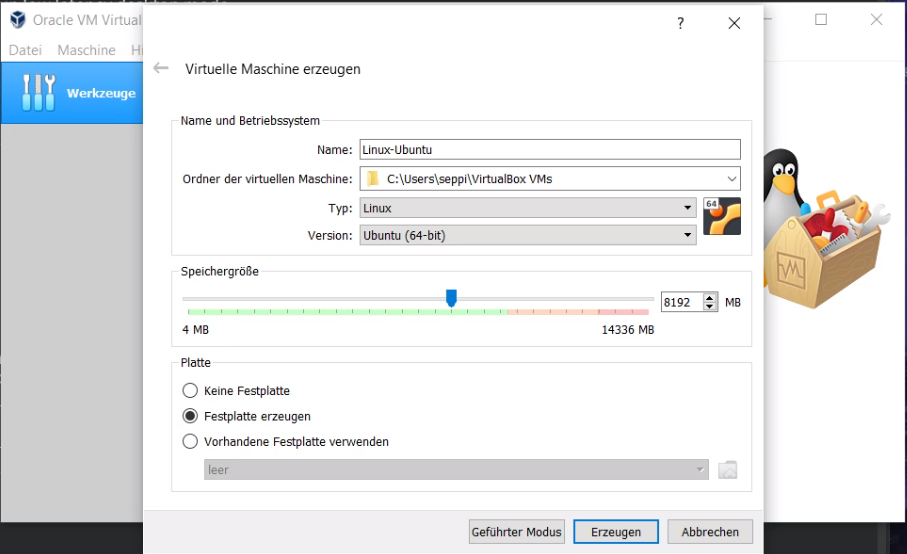
**Betriebssysteme Praktikum 1**

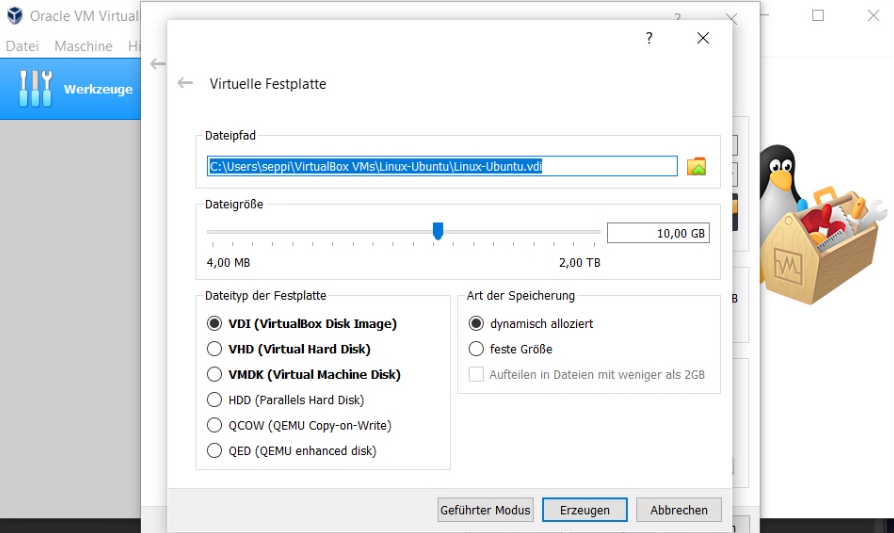
**–**

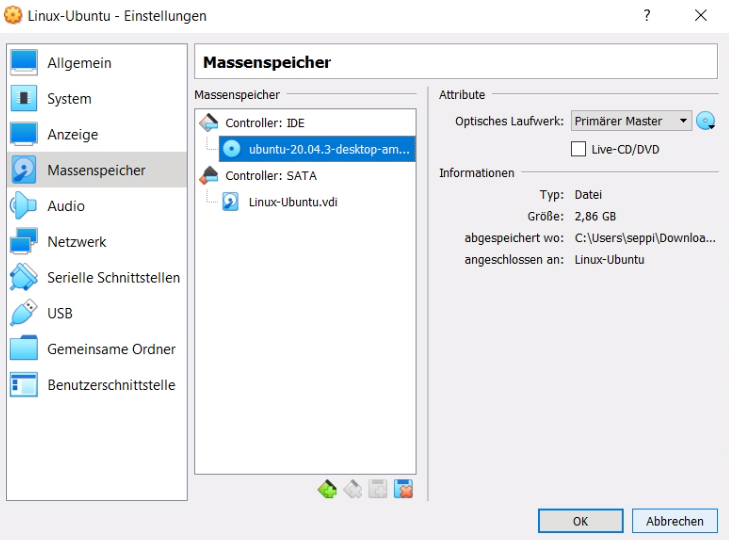
**Jörg Quick 762025 Sebastian Zill 769544**

**Teil 1:**

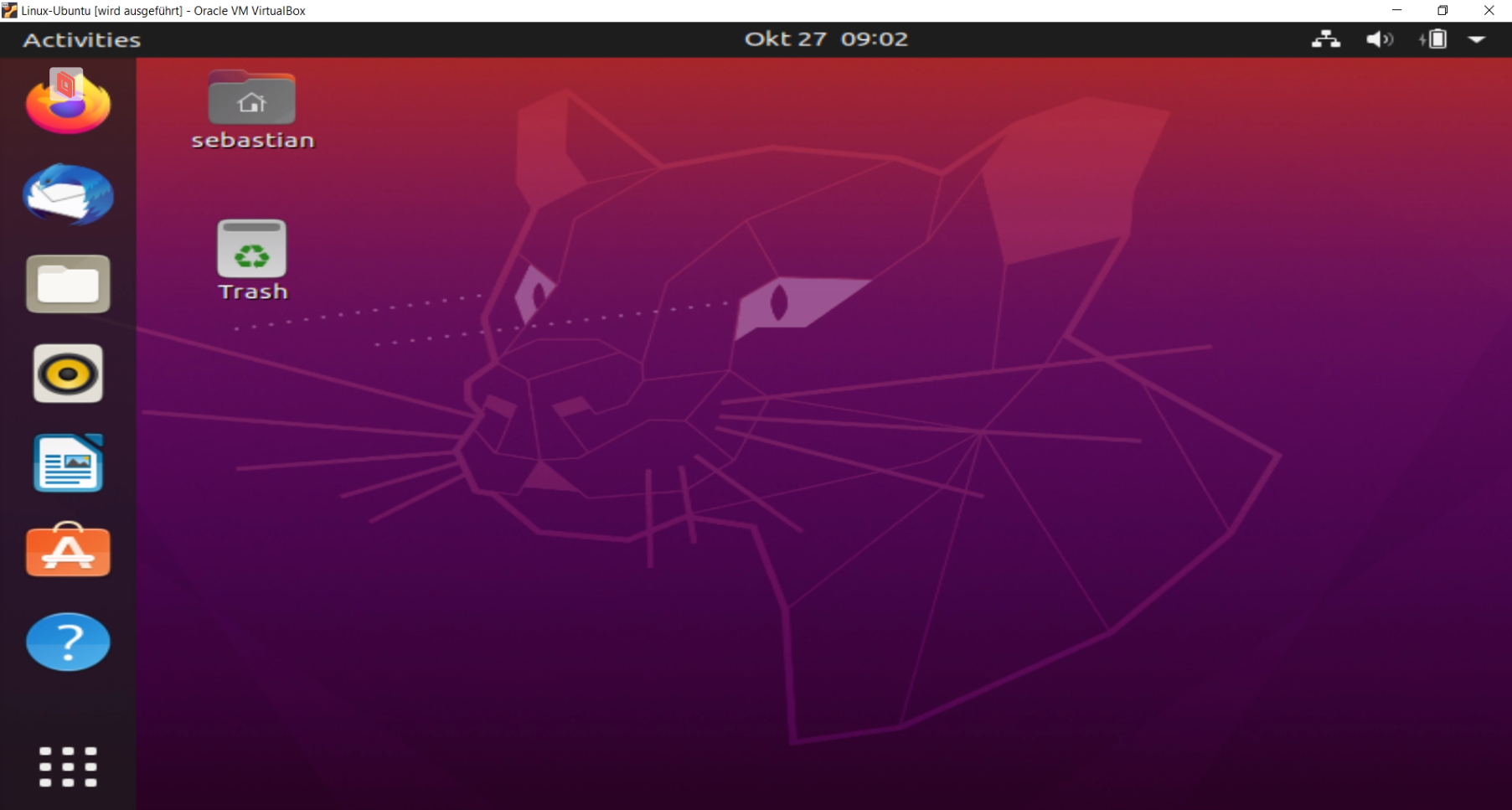
* VM Aufsetzen – Linux Ubuntu über Virtualbox oracle



****

**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 gdp g++ (g++ Compiler install)
* g++ --version (Compiler prüfen)

**Install IDE (Codeblocks)**

* sudo apt-get install codeblocks

**Teil 2.**

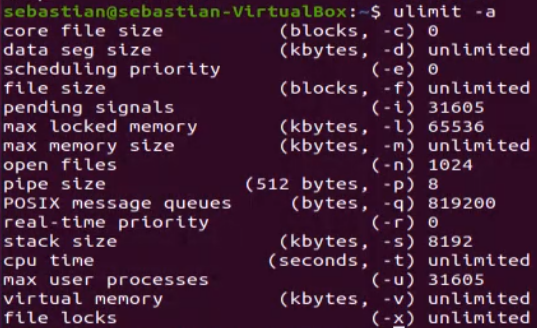
Schreiben Sie eine Funktion **funcMem**, die einen großen Array von Integern zuerst allokiert und dann jedes einzelne Arrayelement anspricht (z.B. durch eine Zuweisung).

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“:



|  |  |  |
| --- | --- | --- |
|  | Dokumentation | Experiment |
| Hauptspeicher | unlimited | Unlimited (Programm läuft bis zu Segmentation-Fehlerfällen) |
| Stackgröße | 8388KB | 3482KB (Segmentation fault) |



1. **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 | - |