

# Operating Systems

## 4 Filesystem and Memory

For the third exercise you have to use python3 as programming language

### 4.1 File Handle

Implement the following pseudocode-functions.

```
procedure write_to_file(filename):
  for i=0 to 25
    print_to_standard_out("Write Line: " + i + actual_timestamp())
    write_to_file(filename, "Write Line: " + i + actual_timestamp())
    wait 1 second

procedure append_to_file(filename):
  for i=0 to 25
    print_to_standard_out("Append Line: " + i + actual_timestamp())
    append_to_file(filename, "Append Line: " + i + actual_timestamp())
    wait 1 second

procedure read_from_file(filename):
  for i=0 to number_of_rows(filename)
    print_to_standard_out(filename_row_number[i])
    wait 1 second

procedure main()
  write_to_file(file1.txt):
  append_to_file(file2.txt):
  read_from_file (file1.txt):
  read_from_file (file2.txt):
```

**Source:** <https://www.digitalocean.com/community/tutorials/how-to-handle-plain-text-files-in-python-3>

What is the **output** of your implementation?

What is a file descriptor?

Where can you find information about the **file descriptor**? (Hint: A **process** is an instance of a computer program that is being executed)

What does **pos**, **flags** and **mnt\_id** mean? (Hint: `/proc/<processid>/fdinfo`)

Let's have a look at the files. What permissions do your files have? (Hint: `"ls -lahi /proc/<processid>/fd/"`)

**For the next 3 Tasks you must install Ubuntu on your own Machine**

1 CPU Core, 2 GB RAM, 32 GB HDD

<https://ubuntu.com/tutorials/how-to-run-ubuntu-desktop-on-a-virtual-machine-using-virtualbox#1-overview> (Linux and Windows)

<https://www.youtube.com/watch?v=v1JVqd8M3Yc> (Linux and Windows)

<https://freegistutorial.com/install-ubuntu-22-10-on-m1-mac/> (Mac)

[https://www.youtube.com/watch?v=EiO\\_CHfSn2s](https://www.youtube.com/watch?v=EiO_CHfSn2s) (Mac)

## 4.2 Run the script every 10 minutes

Use the Cron-Daemon to run the script from 4.1 every 10 minutes.

**Source:** <https://wiki.ubuntuusers.de/Cron/>

What is a cronjob?

## 4.3 While 1

Add an infinite loop to your code.

```
procedure main()
  while(1)
    write_to_file(file1):
    append_to_file(file2):
```

Find out the **pid** (<processid>) of the process and look at `"/proc/<processid>/maps"`. What does the values mean?

Look at `"/proc/<processid>/smaps"`. What does the values **Size**, **Rss**, **Pss**, **Shared\_Clean**, **Shared\_Dirty**, **Private\_Clean**, **Private\_Dirty**, **Referenced**, **Swap** and **SwapPss** mean?

What is the **pagesize** of your system?

How can you print out all **major** and **minor pagefaults**?

Start htop and enter „swapoff -a“ on the terminal. What happens?

Reboot your system and print out all your page faults again.

## 4.4 Fill the RAM

Implement the following python script "ram.py".

```
import sys, time
some_str = ' ' * 1024 * 1024 * 1024 * int(sys.argv[1])
while 1:
    print("true")
    time.sleep(1)
```

Start the python script on your Ubuntu. Start the script with the following arguments: 1, 2, 3, 4 and 5. (e.g.: `python ram.py 1`) What happens?

Start htop and enter „swapoff -a“ on the terminal. Start the script with the following arguments: 1, 2, 3, 4 and 5. What happens now? What has changed?

Try to change your operating system, to run the python script with all arguments (1, 2, 3, 4 and 5).

<https://askubuntu.com/questions/178712/how-to-increase-swap-space>

Compare the `"/proc/meminfo"` or `"htop"` with a running ram.py and a not running ram.py.

What is paging?