OD - LAB 06: Operating Systems

Exercise 1

Your tasks:

1. Make a script that calculates Fib35
    - **Hint**: https://www.geeksforgeeks.org/python-program-for-program-for-fibonacci-numbers-2/
- Make the script print the calculation time.
    - **Hint:** If the calculation is too fast, increase the Fib number, and if the calculation is taking too long, decrease the Fib number.

The exercise will return the following message:

```
[TIME]
```

1. & 2. Creating the python file:

main.py

```
# Function for nth Fibonacci number
import time


def Fibonacci(n):
    # Check if input is 0 then it will
    # print incorrect input
    if n < 0:
        print("Incorrect input")

    # Check if n is 0
    # then it will return 0
    elif n == 0:
        return 0

    # Check if n is 1,2
    # it will return 1
    elif n == 1 or n == 2:
        return 1
```

```
    else:
        return Fibonacci(n - 1) + Fibonacci(n - 2)


# Driver Program
StartTime = time.time()
print(Fibonacci(35))
EndTime = time.time()
print(EndTime - StartTime)
```

Exercise 02

Your tasks:

1. Create a Dockerfile which runs the python script from the previous exercise.
2. Create a Docker image from the Dockerfile.
3. Spin up containers from the image and use different options for CPU.
    a. **Notice** the difference in time it takes for the program to do its calculations.
4. Talk with the other members in your group and explain what is happening?
5. **Hint**: Information can be found here

The exercise will return the following message:

```
[TIME]
```

1. Creating the dockerfile that runs the python script

```
FROM python:alpine3.7
COPY . /app
WORKDIR /app
ENTRYPOINT [ "python" ]
CMD [ "main.py" ]
```

2. Creating the docker image

```
docker build -t exercise2 .
```

3. Spinning up the containers

```
docker run --cpus=0.5 exercise
```

Exercise 3

Your tasks:

1. Create a python file similar to the one in the code snippet above.
2. Create a Dockerfile which runs the python script.
3. Create a Docker image from the Dockerfile.
4. Spin up containers from the image and use different options for memory.
    a. **Notice** the difference in time it takes for the program to do its calculations.

- Talk with the other members in your group and explain what is happening?
    - **Hint**: Information can be found here

1. Crating the python file

```
# Function for nth Fibonacci number
import time


def Fibonacci(n):
    # Check if input is 0 then it will
    # print incorrect input
    if n < 0:
        print("Incorrect input")

    # Check if n is 0
    # then it will return 0
    elif n == 0:
        return 0

    # Check if n is 1,2
    # it will return 1
```

```
    elif n == 1 or n == 2:
        return 1

    else:
        return Fibonacci(n - 1) + Fibonacci(n - 2)

# Method which uses a lot of space
def use_space(loops):
    for i in range(0, loops):
        arr = bytearray(512000000)

# Program
start_time = int(round(time.time() * 1000))
use_space(2)
end_time = int(round(time.time() * 1000))
print('time: ' + str(end_time - start_time))
```

2. Creating the dockerfile

```
FROM python:alpine3.7
COPY . /app
WORKDIR /app
ENTRYPOINT [ "python" ]
CMD [ "exercise3.py" ]
```

3. Creating the docker image

```
docker build -t exercise3 .
```

4. Spinning up the image

```
docker run --memory=1000m exercise3
```