

Nama: Diana Sinthya Putri
NPM: 21083010090
Kelas: Sistem Operasi B

TUGAS

getpid digunakan untuk mengambil ID proses

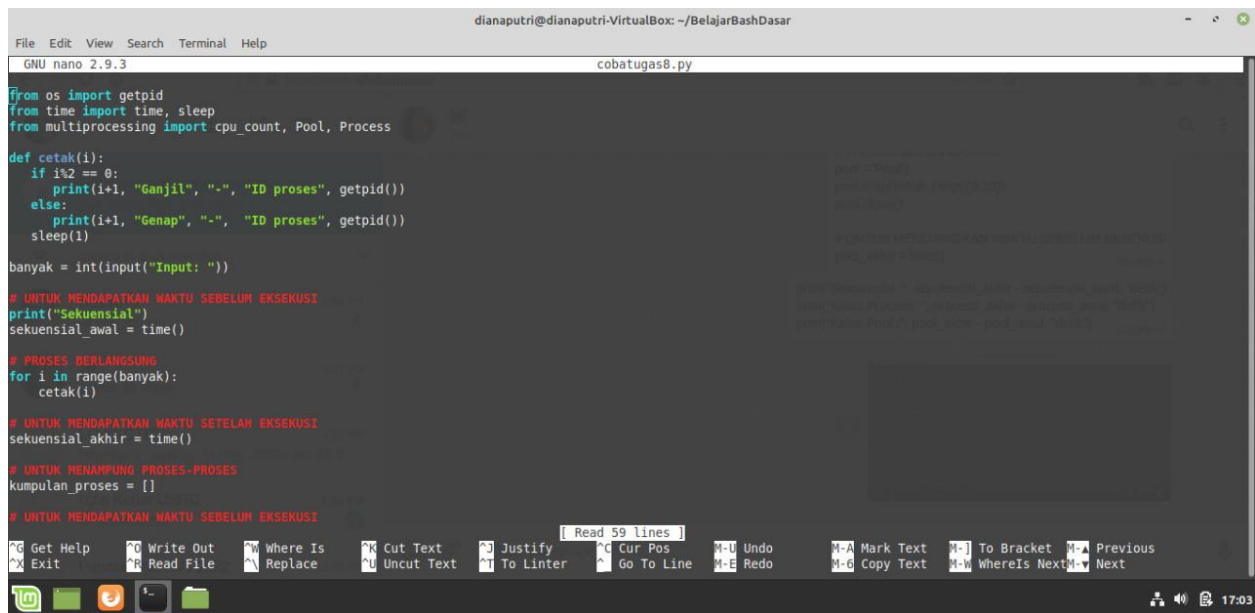
time digunakan untuk mengambil waktu(detik)

sleep digunakan untuk memberi jeda waktu(detik)

cpu_count digunakan untuk melihat jumlah CPU

Pool adalah sebuah class pada library multiprocessing yang digunakan untuk melakukan pemrosesan paralel dengan menggunakan proses sebanyak jumlah CPU pada komputer

Process adalah sebuah class pada library multiprocessing yang digunakan untuk melakukan pemrosesan paralel dengan menggunakan proses secara beruntun pada komputer



```
File Edit View Search Terminal Help
GNU nano 2.9.3 cobatugas8.py

from os import getpid
from time import time, sleep
from multiprocessing import cpu_count, Pool, Process

def cetak(i):
    if i%2 == 0:
        print(i+1, "Ganjil", "-", "ID proses", getpid())
    else:
        print(i+1, "Genap", "-", "ID proses", getpid())
    sleep(1)

banyak = int(input("Input: "))

# UNTUK MENDAPATKAN WAKTU SEBELUM EKSEKUSI
print("Sekuensial")
sekuensial_awal = time()

# PROSES BERLANGSUNG
for i in range(banyak):
    cetak(i)

# UNTUK MENDAPATKAN WAKTU SETELAH EKSEKUSI
sekuensial_akhir = time()

# UNTUK MENAMPUNG PROSES-PROSES
kumpulan_proses = []

# UNTUK MENDAPATKAN WAKTU SEBELUM EKSEKUSI
```

Dapat diperhatikan dengan seksama bahwa ID proses tiap memanggil fungsi cetak adalah berbeda-beda. Ini menandakan bahwa tiap pemanggilan fungsi cetak ditangani oleh satu proses saja. Kemudian untuk pemanggilan selanjutnya ditangani oleh proses yang lain.

Kumpulan proses harus ditampung dan digabung menjadi satu(`p.join()`) agar tidak merambah ke proses selanjutnya. Silahkan eksekusi file berikut pada terminal anda, maka anda akan paham apa yang saya maksudkan.


```
dianaputri@dianaputri-VirtualBox:~/BelajarBashDasar$ python3 cobatugas8.py
Input: 3
Sekuensial
1 Ganjil - ID proses 23584
2 Genap - ID proses 23584
3 Ganjil - ID proses 23584
multiprocessing.Process
1 Ganjil - ID proses 23585
2 Genap - ID proses 23586
3 Ganjil - ID proses 23587
multiprocessing.Pool
1 Ganjil - ID proses 23588
2 Genap - ID proses 23588
3 Ganjil - ID proses 23588
Waktu eksekusi sekuensial : 3.0021708011627197 detik
Waktu eksekusi kelas Process : 1.0369369983673096 detik
Waktu eksekusi kelas Pool : 3.0570435523986816 detik
dianaputri@dianaputri-VirtualBox:~/BelajarBashDasar$
```

Sudah sewajarnya proses sekuensial lebih lambat dibanding multiprocessing namun bukan berarti kita harus melakukan multiprocessing terus menerus, gunakan metode sesuai kebutuhan. Nah apabila barisan kode di atas dikumpulkan jadi satu maka jadinya akan seperti ini.