

Nama : Fadhilah Nur Hidayat

NPM : 21083010082

Kelas : Sistem Operasi C

---

Pertama yang dilakukan adalah membuat file .py dengan cara nano namafilename.py.

```
fadhil@fadhil-VirtualBox:~$ nano tugas-8.py
```

Setelah itu, menuliskan script python yang sesuai dengan soal yang telah diberikan. Sehingga input dari script python tersebut adalah sebagai berikut :

```
GNU nano 6.2 tugas-8.py
from os import getpid
from time import time,sleep
from multiprocessing import cpu_count, Pool, Process

def cetak(i):
    if (i+1)%2==0:
        print(i+1, "genap - ID Process", getpid())
    else:
        print(i+1, "ganjil - ID Process", getpid())
        sleep(1)

n=int(input("Angka batasan? "))

#SEKUENSIAL
sekuensial_awal = time()
print("Sekuensial")
for i in range(n):
    cetak(i)
sekuensial_akhir=time()
```

- a. Import modul yang diperlukan.  
getpid digunakan untuk mendapatkan proses id, time digunakan untuk mengambil waktu pada proses dijalankan atau diakhiri, sleep digunakan untuk menunda waktu eksekusi, cpu\_count digunakan untuk menghitung berapa banyak cpu yang tersedia, Pool digunakan untuk melakukan pemrosesan parallel dengan menggunakan proses sebanyak jumlah CPU pada computer. Process digunakan untuk melakukan pemrosesan parallel dengan menggunakan proses secara beruntun pada computer.
- b. gunakan sekuensial processing.
  - sekuensial\_awal dan sekuensial\_akhir adalah variable untuk menyimpan waktu durasi selama proses sekuensial processing berlangsung.
  - lakukan Looping sebanyak angka yang dimasukkan oleh user, dan gunakan fungsi cetak yang sudah terisi di awal untuk mencetak setiap angka ganjil atau genap dengan proses idnya masing – masing.

```

#MULTIPROCESSING DENGAN KELAS PROCESS
process_awal=time()
print("Multiprocess.process")
for i in range(n):
    p=Process(target=cetak, args=(i, ))
    p.start()
    p.join()
process_akhir=time()

#MULTIPROCESSING DENGAN KELAS POOL
pool_awal=time()
pool = Pool()
print("Multiprocess.pool")
pool.map(cetak,range(0,n))
pool.close()
pool_akhir=time()

#BANDINGKAN WAKTU EKSEKUSI
print("Perbandingan waktu")
print("Sekuensial:", sekuensial_akhir - sekuensial_awal, "detik")
print("Kelas Process:", process_akhir - process_awal, "detik")
print("Kelas Pool:", pool_akhir - pool_awal, "detik")

```

- c. gunakan multiprocessing dengan kelas process.  
process\_awal dan process\_akhir adalah variable menyimpan waktu awal mulainya proses dijalankan dan waktu berakhirnya proses.
- d. gunakan multiprocessing dengan kelas pool.  
pool\_awal dan pool\_akhir adalah variable menyimpan waktu awal mulainya proses dijalankan dan waktu berakhirnya proses.
- e. Kemudian, bandingkan setiap jenis eksekusi dengan waktu akhir – waktu awal untuk melihat berapa lama pemrosesan berlangsung.

Setelah menyimpan file script tersebut, maka kita dapat menjalankan program dengan mengetikkan python3 tugas-8.py. Sehingga output yang muncul adalah sebagai berikut :

```

fadhil@fadhil-VirtualBox:~$ python3 tugas-8.py
Angka batasan? 3
Sekuensial
1 ganjil - ID Process 2327
2 genap - ID Process 2327
3 ganjil - ID Process 2327
Multiprocess.process
1 ganjil - ID Process 2328
2 genap - ID Process 2329
3 ganjil - ID Process 2330
Multiprocess.pool
1 ganjil - ID Process 2331
2 genap - ID Process 2331
3 ganjil - ID Process 2331
Perbandingan waktu
Sekuensial: 3.004215717315674 detik
Kelas Process: 3.0256776809692383 detik
Kelas Pool: 3.0658740997314453 detik

```

Pada program tersebut menjelaskan bahwa, terdapat 3 bentuk yaitu sekuensial, multiprocessing, dan multiprocessing.pool. Dimana ketiganya akan menghasilkan angka masing-masing yang berbeda sesuai dengan system atau script yang telah dibuat. Selain itu, program juga menampilkan waktu eksekusi dari system / program ketika user telah menginputkan angka.