

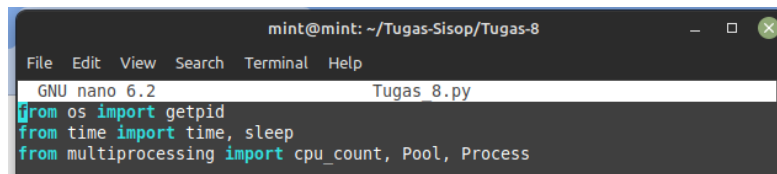
Tugas Pertemuan 8 Sistem Operasi

Nama : Muhammad Rafli

NPM : 21083010066

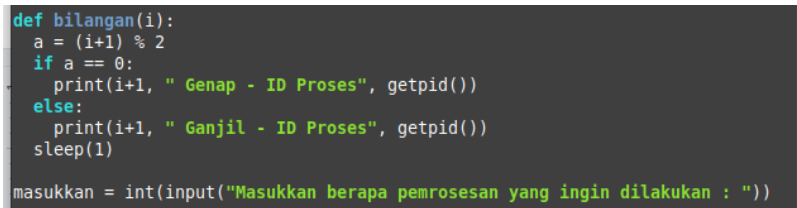
Latihan Soal

Pada tugas kali ini diperintahkan untuk membuat sebuah pemrosesan sekuensial dan paralel untuk menentukan sebuah bilangan itu genap atau ganjil. Dengan syarat nilai pada argumen sleep yaitu 1 detik, terdapat input sebuah bilangan bulat untuk menentukan batas perulangan, dan terakhir ditampilkan waktu dalam pemrosesan sekuensial dan paralel.



```
mint@mint: ~/Tugas-Sisop/Tugas-8
File Edit View Search Terminal Help
GNU nano 6.2 Tugas_8.py
from os import getpid
from time import time, sleep
from multiprocessing import cpu_count, Pool, Process
```

Pertama mengimport library yang kita perlukan dalam menjalankan pemrosesan sekuensial dan paralel. Library yang diperlukan dapat dilihat pada gambar diatas.



```
def bilangan(i):
    a = (i+1) % 2
    if a == 0:
        print(i+1, " Genap - ID Proses", getpid())
    else:
        print(i+1, " Ganjil - ID Proses", getpid())
    sleep(1)

masukkan = int(input("Masukkan berapa pemrosesan yang ingin dilakukan : "))
```

Lalu dilanjutkan membuat fungsi yang digunakan untuk menghitung sebuah bilangan merupakan bilangan ganjil atau genap. Dalam fungsi tersebut terdapat variabel yang berisi perhitungan modulus angka yang dimasukkan terhadap 2. Bilangan ditambahkan 1 karena pada perulangannya dimulai dari 0. Lalu bila modulusnya sama dengan 0 berarti bilangan tersebut genap dan bila tidak maka bilangan tersebut merupakan ganjil. Id proses juga akan ditampilkan beserta ganjil genap bilangan. Dalam akhir fungsi juga dimasukkan fungsi sleep dengan nilainya adalah 1 yang berarti 1 detik. Kemudian dibuat variabel masukkan untuk menginputkan batas bilangan yang ingin dilakukan pemrosesan berupa integer.



```
print("\nSekuenisial")
sekuensial_awal = time()

for i in range(masukkan):
    bilangan(i)

sekuensial_akhir = time()
```

Pada proses sekuensial pertama dibuat variabel untuk menampung waktu awalnya. Lalu dilakukan proses looping dengan batas variabel masukkan yang sebelumnya sudah dijelaskan dan dipanggil fungsi bilangan untuk menghitung bilangan tersebut ganjil atau genap. Terakhir dibuat variabel lagi untuk menghitung waktu setelah proses berakhir.

```

print("\nMultiprocessing.Process")
kumpulan_proses = []

process_awal = time()

for i in range(masukkan):
    p = Process(target=bilangan, args=(i,))
    kumpulan_proses.append(p)
    p.start()

for i in kumpulan_proses:
    p.join()

process_akhir = time()

```

Selanjutnya adalah pemrosesan multiprocessing dengan kelas process. Pertama dibuat sebuah wadah untuk menampung semua pemanggilan yang ditangani oleh proses yang berbeda. Lalu buat variabel untuk waktu awal program berjalan. Proses looping dijalankan sesuai dengan pemrosesan paralel dan nantinya hasil dari tiap prosesnya ditambahkan pada wadah yang telah dibuat diawal. Lalu dilakukan looping lagi untuk menampung kumpulan proses dan digabungkan agar tidak merambah ke proses selanjutnya. Terakhir variabel untuk menentukan waktu selesainya proses berakhir.

```

print("\nMultiprocessing.Pool")
pool_awal = time()

pool = Pool()
pool.map(bilangan, range(0,masukkan))
pool.close()

pool_akhir = time()

```

Terakhir multiprocessing dengan menggunakan pool. Pertama kita buat sebuah variabel untuk menghitung waktu dimulainya program. Lalu proses paralel pool. Pada proses tersebut nantinya output dibagi ke keempat CPU untuk diproses. Terakhir yaitu variabel untuk menentukan berakhirnya program tersebut dijalankan.

```

print("\nWaktu eksekusi sekuensial :", sekuensial_akhir - sekuensial_awal, "detik")
print("Waktu eksekusi multiprocessing.Process :", process_akhir - process_awal, "detik")
print("Waktu eksekusi multiprocessing.Pool :", pool_akhir - pool_awal, "detik")

```

Selanjutnya membuat hasil dari waktu yang telah dilakukan sebelumnya. Dibuat sebuah print out string lalu untuk menghitung waktunya yaitu dengan mengurangkan waktu akhir dari proses dikurang dengan waktu awal proses dimulai.

```
mint@mint: ~/Tugas-Sisop/Tugas-8
File Edit View Search Terminal Help
mint@mint:~/Tugas-Sisop/Tugas-8$ nano Tugas_8.py
mint@mint:~/Tugas-Sisop/Tugas-8$ python3 Tugas_8.py
Masukkan berapa pemrosesan yang ingin dilakukan : 3

Sekuensial
1 Ganjil - ID Proses 26706
2 Genap - ID Proses 26706
3 Ganjil - ID Proses 26706

Multiprocessing.Process
1 Ganjil - ID Proses 26707
2 Genap - ID Proses 26708
3 Ganjil - ID Proses 26709

Multiprocessing.Pool
1 Ganjil - ID Proses 26710
2 Genap - ID Proses 26711
3 Ganjil - ID Proses 26712

Waktu eksekusi sekuensial : 3.003237247467041 detik
Waktu eksekusi multiprocessing.Process : 1.0167827606201172 detik
Waktu eksekusi multiprocessing.Pool : 1.04646635055542 detik
mint@mint:~/Tugas-Sisop/Tugas-8$
```

Lalu didapatkan output seperti pada gambar tersebut dimana inputan yang dimasukkan sebagai batas dari proses loopingnya adalah 3. Didapat ganjil genap dari bilangan berurutan dari 1 hingga 3 sesuai dengan inputan yang telah dimasukkan. Dapat dilihat Id dari proses sekuensial itu sama semua dan Id dari proses paralel berbeda - beda artinya pada proses sekuensial program dijalankan satu - persatu sedangkan pada paralel dikerjakan secara bersamaan. Lalu terdapat waktu dari pemrosesan tersebut dan pada proses sekuensial diperlukan waktu yang lebih lama daripada eksekusi yang dilakukan oleh pemrosesan paralel.