



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71230976
Nama Lengkap	Galih Pramana Chandra Prasetya
Minggu ke / Materi	08 / Membaca dan Menulis File

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2024

BAGIAN 1: MATERI MINGGU INI (40%)

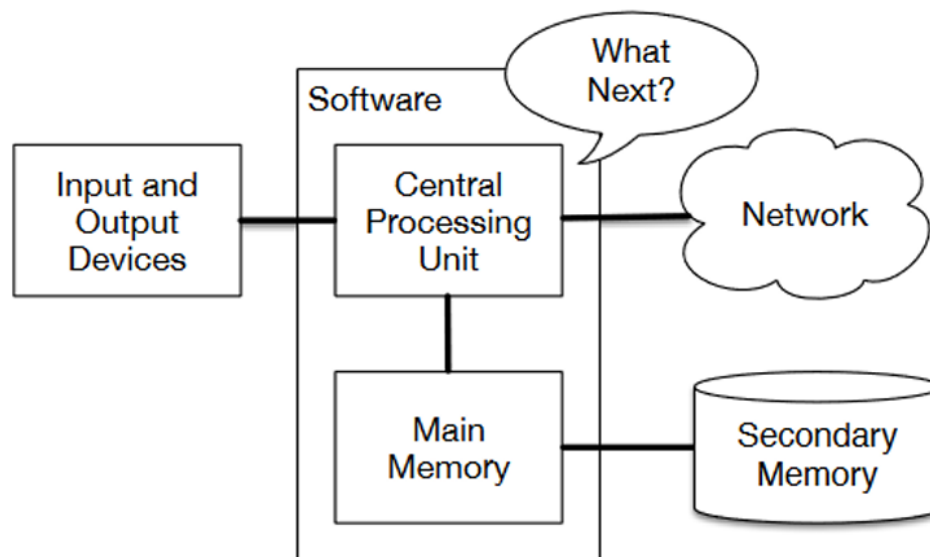
Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

Pengantar File

Program yang berjalan membutuhkan memory primer di dalam computer. Semua data yang ada di program tersebut setelah dijalankan dan dimatikan, maka semua data di dalam program tersebut juga ikut hilang atau bersifat tidak permanen (volatile).

Agar bisa menyimpan data, program harus disimpan dalam secondary memory. File yang disimpan pada secondary memory tidak akan hilang walaupun computer dimatikan.

File pada dasarnya adalah bit-bit yang saling berelasi satu sama lain sebagai satu kesatuan. File bisa berupa file sistem, file program, file multimedia, file teks, dll. File memiliki property seperti nama file, ukuran, letak hardisk, owner, hak akses, dll.



Gambar 8.1: Secondary Memory di Komputer (diambil dari modul)

Pengaksesan File

Langkah-langkah untuk dapat mengakses file :

1. Menyiapkan file dan path yang akan diakses
2. Open file

3. Lakukan sesuatu dengan file tersebut, seperti ditampilkan isinya (read) atau dirubah/ditulis (write)
4. Close file

Dalam python dapat dibuat seperti berikut:

1. Menyiapkan file txt dengan nama "namafile".
2. Open file dengan cara berikut ini:

```
1 handle = open('nama file.txt')
2 print(handle)
```

Hasilnya:

```
<_io.TextIOWrapper name='namafile.txt' mode='r' encoding='cp1252'>
```

Hasilnya berupa tampilan nama file, modenya, dan encoding yang digunakan yaitu Unicode cp1252 dari sistem io pada python.

Jika nama file tidak ada / tidak ditemukan, maka output akan error:

```
Traceback (most recent call last):
  File "d:\Pra Alpro sem 2\pra alpro\Laprak 08\laprak08.py", line 1, in <module>
    handle = open('file.txt')
             ^^^^^^^^^^^^^^^^^
FileNotFoundError: [Errno 2] No such file or directory: 'file.txt'
```

Pada kasus diatas terjadi error karena file yang ingin di akses tidak terdapat didalam folder file yang benar atau terdapat kesalahan dalam penulisan nama file.

Cara pembacaan file teks biasanya menggunakan model baca baris demi baris untuk setiap string yang diteminukan sampai dengan EOF (End of File). Contoh tampilan baris-baris dalam sebuah file:

3. Lakukan sesuatu dengan file tersebut, seperti ditampilkan isinya (read) atau dirubah/ditulis (write)
 - Read: program hanya dapat membaca isi dari file, namun tidak dapat memanipulasi.
 - Write: program dapat memanipulasi isi file, namun hanya berlaku sekali. Jika program write dijalankan lagi, maka data dari jalan sebelumnya akan hilang digantikan dengan yang baru.
 - Append: program dapat memanipulasi file dengan menambahkan isi data baru. Data lama tidak akan hilang karena program hanya akan menambahkan data yang baru di bawah data lama.

-
- 4. Close File

Untuk menutup file bisa menggunakan cara berikut ini

```
4 handle.close()
```

Manipulasi File

Sebuah program sederhana dapat memanipulasi file dengan metode membaca, mengekstrak bagian dari file, dan sejenisnya. Proses manipulasi file dimulai dengan mengakses file. Selanjutnya, program memerlukan metode akses yang diinginkan: apakah itu membaca (read), menulis (write), atau menambahkan (append). Jika tidak ada spesifikasi metode, maka secara default program akan menggunakan metode membaca (read). Terakhir, berkas ditutup untuk menghentikan akses program terhadap berkas tersebut.

Berikut ini contoh menggunakan perulangan:

```
handle = open('namafile.txt')
count = 0
for line in handle:

    count = count + 1
    print('Line Count:', count)
```

Berikut ini contoh menggunakan metode string:

```
handle = open('namafile.txt')
hasil = handle.read()

print("Ukuran: " + len(hasil) + "bytes")
print("Huruf dari belakang sendiri mundur 16 huruf adalah: " + hasil[-16::1])
```

Contoh lain menggunakan startswith:

```
handle = open('namafile.txt')
count = 1

for line in handle:
    if line.startswith("Date:") and count <= 10:
        count += 1
        print(line)
```

Masih banyak lagi contoh penggunaan metode string maupun list maupun tuple untuk memanipulasi file.

Penyimpanan File

Sebuah program sederhana juga dapat, menghapus, menambah serta mengganti isi dari sebuah file menggunakan metode write maupun append. Terdapat perbedaan antara write dan append meskipun memiliki kegunaan yang sama. Fungsi Write memungkinkan penulisan data baru ke dalam file, namun dengan catatan bahwa data yang ada sebelumnya akan terhapus atau ditimpa oleh data yang baru, sehingga data lama hilang dan terhapus. Sedangkan append menambahkan data baru ke dalam file, sehingga data lama dalam file masih tersimpan.

1. **Write** : Menggunakan “w” dalam perintah open() .

```
file = open('namafile.txt', "w")

teks= "teks ini akan masuk ke dalam file namafile.txt\n"

file.write(teks)

file.close
```

metode ini akan merubah keseluruhan file atau menghapus seluruh isi file dan menggantikannya dengan teks yang baru.

2. **Append**: Menggunakan “a” dalam perintah open().

```
file = open('namafile.txt', "a")

teks= "teks ini akan masuk ke dalam file namafile.txt\n"

file.write(teks)

file.close
```

hasilnya akan menambahkan teks tanpa menghapus teks yang sudah ada

Kegiatan Praktikum

1. Kasus 8.1 (Menampilkan url dalam sebuah file)

```
filename = input("nama file: ")
handle = open(filename)
c = 0

for line in handle:
    if line.find("ac.uk") != -1:
        c+= 1
```

```
        print("Web domain 'ac.uk' ditemukan di \"" + line.strip() + "\"")
print("Jumlah: ",c)
```

2. Kasus 8.2 (menampilkan baris baris file yang diawali string 'Subject:')

```
filename = input("nama file: ")
handle = open(filename)
c = 0

for line in handle:
    if line.startswith("Subject:"):
        c += 1
        line = line.strip().title()

        print(line)

print("Jumlah: ",c)
```

3. Kasus 8.3 (menampilkan ukuran file dalam KB dari sebuah file teks dan menghandle error jika file yang diinputkan tidak ditemukan.)

```
nama = input("nama file: ")
try:
    file = open(nama)
    total = 0
    for baris in file:
        total += len(baris)
        kb = total / 1000
    print("Ukuran: " + (kb) + " KB")

except:
    print("File tidak ditemukan!")
```

BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

SOAL 1

Source code:

```
#Latihan Mandiri 8.1
def banding_teks():
    file1 = input("Input file pertama: ")
    file2 = input("Input file kedua: ")

    with open(file1, 'r') as f1, open(file2, 'r') as f2:
        lines1 = f1.readlines()
        lines2 = f2.readlines()

    x = max(len(lines1), len(lines2))

    for i in range(x):
        if lines1[i] != lines2[i]:
            print(f"Perbedaan ada di {i+1}:")
            print(f"  File 1: {lines1[i].strip()}")
            print(f"  File 2: {lines2[i].strip()}")

banding_teks()
```

Output:

```
PS D:\Pra Alpro sem 2\pra alpro\Laprak 08> & C:/Users/Asus/A
Input file pertama: teks1.txt
Input file kedua: teks2.txt
Perbedaan ada di 1:
  File 1: galih
  File 2: rainie
Perbedaan ada di 2:
  File 1: rainie
  File 2: galih
Perbedaan ada di 3:
  File 1: haloo
  File 2: haii
```

Penjelasan:

- Pengguna diminta untuk memasukkan nama dua file yang ingin dibandingkan.
- File pertama dan kedua dibuka secara bergantian.
- Isi dari setiap file dibaca dan disimpan.
- Perbandingan dilakukan untuk setiap baris pada kedua file.
- Jika terdapat perbedaan pada baris yang sama, pesan perbedaan beserta isi baris dari masing-masing file dicetak.
- Proses ini diulangi untuk setiap baris hingga selesai.
- Setelah selesai, file-file yang dibuka akan secara otomatis ditutup.
- Fungsi `banding_teks()` dipanggil untuk menjalankan seluruh proses tersebut.

SOAL 2

Source Code:

```
#LATIHAN MANDIRI 8.2
with open("soal.txt", "r") as pertanyaan:
    for index, baris in enumerate(pertanyaan, start=1):
        i = baris.strip().split(" || ")
        print(f"{i[0]}")
        jawab = input("Jawab: ")
        if jawab.lower() == i[1].lower():
            print("Jawaban Benar")
        else:
            print("Jawaban Salah")
```

Output:

```
PS D:\Pra Alpro sem 2\pra alpro\Laparak 08> & C:/Us
1+1 =
Jawab: 2
Jawaban Benar
Bendera Indonesia?
Jawab: Merah Putih
Jawaban Benar
Kota gudeg adalah:
Jawab: Solo
Jawaban Salah
Komponen PC untuk penyimpanan file adalah...
Jawab: Harddisk
Jawaban Benar
50 * 20 =
Jawab: 10
Jawaban Salah
```

Penjelasan:

- Berkas teks "soal.txt" dibuka untuk dibaca menggunakan pernyataan with open() dengan mode "r" (baca).
- Dalam loop for, setiap baris dari file teks dibaca dan diproses secara bergantian.
- Fungsi enumerate() digunakan untuk menghasilkan indeks dan nilai untuk setiap baris dalam file, dimulai dari 1 (start=1).
- Setiap baris dipisahkan menjadi dua bagian menggunakan split(" || "), sehingga mendapatkan pertanyaan dan jawaban yang dipisahkan oleh " || ".

- Pertanyaan (bagian pertama) dicetak ke layar.
- Pengguna diminta untuk memberikan jawaban.
- Jawaban yang diberikan dibandingkan dengan jawaban yang seharusnya (bagian kedua) dari baris yang bersangkutan.
- Jika jawaban pengguna sama dengan jawaban yang seharusnya (ignorasi kapitalisasi), pesan "Jawaban Benar" dicetak.
- Jika jawaban pengguna tidak sama dengan jawaban yang seharusnya, pesan "Jawaban Salah" dicetak.
- Proses ini berulang sampai semua baris dari berkas teks diproses.

Link GitHub: <https://github.com/GalihPramana/Praktikum-Alpro-71230976.git>