

FILE DATA

OBJEKTIF :

1. Mahasiswa Mampu Memahami Tentang File Data pada Bahasa C.
-

1.1 FILE DATA

Input atau output functions menangani data pada perangkat penyimpanan sekunder seperti hardisk dan C dapat menangani files dengan dua cara yaitu sebagai stream-oriented data files dan system oriented data files.

1. Stream oriented data adalah data disimpan dengan cara yang sama seperti yang terlihat di layer. Jadi, operasi input dan output seperti konversi data berlangsung otomatis.
2. System oriented data adalah file data lebih terkait pada sistem operasi dan data tersimpan dalam memori tanpa terkonversi formatnya menjadi text.

Terdapat beberapa operasi besar pada file, diantaranya :

1. Create : Membuat sebuah file baru.
2. Opening File : Membuka file yang telah ada.
3. Read : Membaca file yang telah ada.
4. Write : Menuliskan sesuatu ke dalam file.
5. Closing File : Menutup File

Langkah – Langkah Operasi File

1. Deklarasikan variabel file pointer
2. Buka file menggunakan function fopen()
3. Proses file menggunakan function yang cocok
4. Tutup file menggunakan function fclose()

Terdapat banyak function yang dapat digunakan ketika melakukan operasi file. Untuk melakukan file handling pada bahasa C, dan melakukan function input dan output yang terdapat dalam stdio library diantaranya :

1. fopen untuk membuka file
2. fclose untuk menutup file
3. getc untuk membaca karakter yang terdapat dalam file
4. putc untuk menulis karakter pada file
5. getw digunakan untuk membaca integer pada file
6. putw digunakan untuk menulis integer pada file
7. fprintf digunakan untuk mencetak string yang di format ke file
8. fscanf digunakan untuk membaca input yang di format dari file
9. fgets digunakan untuk membaca string dari karakter dari file
10. fputs digunakan untuk menginput string ke dalam sebuah file
11. feof berfungsi mendeteksi akhir dari file

Contoh :

```
#include<stdio.h>
#include<stdlib.h>
int main() {
    FILE *fp; //deklarasi file pointer
    fp = fopen("filename.txt","w"); //membuka file
    //proses
    fprintf(fp,"%s","HELLO WORLD");
    //akhir proses
    fclose(fp); //menutup file
    return 0; }
```

RANGKUMAN

1. Input atau output functions menangani data pada perangkat penyimpanan sekunder seperti hardisk dan C dapat menangani files dengan dua cara yaitu sebagai stream-oriented data files dan system oriented data files.
2. Terdapat dua cara untuk memanggil function argument atau parameter, yaitu dengan dipanggil berdasarkan value dan dipanggil berdasarkan reference.

TUNTUNAN LATIHAN

Berikut ini diberikan contoh program untuk membuat file baru jika sudah ada datanya (overwrite) :

- (a) Mendeklarasikan pre-processor include diikuti dengan file judul atau header file. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  _
```

- (b) Mendeklarasikan fungsi main. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
4  int main()
5  {
```

- (c) Mendeklarasikan variabel nama dengan tipe data char dan variabel umur dengan tipe data integer, kemudian membuat format masukan dengan menggunakan printf dan scanf. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
6  char nama[100];
7  int umur;
8  printf("Masukkan nama : "); scanf("%s", &nama); fflush(stdin);
9  printf("Masukkan umur : "); scanf("%d", &umur); fflush(stdin);
10
```

(d) Membuat perintah untuk menyimpan data ke file dengan menggunakan perintah fprintf. Caranya mengetikkan langsung pada lembar kerja, yaitu :

```
11 FILE | *out=fopen("test.txt", "w");  
12 fprintf(out, "%s#%d\n", nama, umur);  
13 fclose(out);  
14 printf("Sukses menambah data.");  
15 getchar();  
16     return 0;  
17 }
```

LATIHAN

1. Buatlah program untuk membuat append file (menambahkan data pada file baris terakhir. Jika belum ada data/filenya, maka append akan membuatkan file baru.) !