

Operasi-operasi terhadap “file” menggunakan Turbo C++ / TCWin 4.5 beserta contoh program.

MK : Dasar-dasar pemrograman (DDP) praktik tingkat 1 JTK-POLBAN.

Dosen : SN

1. Operasi untuk Membuka File : `fopen()` .

Jenis operasi file (parameter ke-2 pada fungsi `fopen()`) :

- r : menyatakan file hanya akan dibaca.
- w : menyatakan file baru diciptakan.
- a : untuk membuka file yang sudah ada pada disk dan operasi yang akan dilakukan adalah operasi penambahan data pada file (***append***) .
- r+ : untuk membuka file yang sudah ada dan operasi yang akan dilakukan adalah pembacaan dan penulisan
- w+ : untuk membuka file dengan tujuan pembacaan dan penulisan, tetapi jika file sudah ada maka isinya akan dihapus (overwrite)
- a+ : untuk membuka file, dengan operasi yang dapat dilakukan adalah perekaman dan pembacaan. Jika file sudah ada, isinya tidak akan dihapus.

Pada saat file dibuka, file bisa diperlakukan sebagai **file biner** atau sebagai **file teks**.

File biner adalah file yang pola penyimpanan di dalam disk adalah dalam bentuk biner, yaitu seperti bentuk pada memori (RAM). Misalnya data bertipe integer selalu akan menempati ruang 2 byte, berapapun harganya.

Sedangkan file teks merupakan file yang pola penyimpanan datanya dalam bentuk karakter. Bilangan bertipe integer pada file teks bisa saja menempati ruang 1 byte, 2 byte, atau lebih, tergantung nilai bilangan. Cth bilangan 54 akan disimpan dalam 2 byte (berupa karakter 5 dan 4), sedangkan bilangan 123 akan disimpan dalam 3 byte. File teks seperti ini akan bisa dilihat isinya menggunakan perintah TYPE dalam DOS prompt atau menggunakan text editor (misal notepad atau text editor lainnya).

Penambahan yang perlu dilakukan untuk menentukan mode teks atau biner berupa :

- t : untuk mode teks

b : untuk mode biner

Contoh :

"rt" berarti mode file adalah teks dan file hendak dibaca.

"rt+" berarti mode file adalah teks dan file bisa dibaca atau ditulisi.

"rb" berarti mode file adalah biner dan file hendak dibaca .

Contoh potongan program :

```
pf=fopen( "BILANGAN.DAT" , "rb" );
```

Parameter rb berarti untuk membaca file bertipe biner;

```
pf=fopen( "BILANGAN.DAT" , "wb" );
```

Parameter wb menunjukkan bila file belum ada, maka akan dibuat.

2. Operasi Baca Tulis (I/O) ke File Teks.

Note : Khusus mengenai fungsi *fopen()*, *fgets()*.

Operasi READ terhadap file teks menggunakan fungsi *fgets()*. Perilaku fungsi ini adalah : baca string dari file teks yang dibukan menggunakan fungsi *fopen()* selama masih ada isinya.

Contoh program dengan TCWin 4.5 terlampir untuk membaca isi file teks adalah : **fgets_wi.cpp**, berikut contoh file teksnya : **huruf.txt** dan **nama.txt**.

Program ini membaca isi file teks lalu menampilkan hasil pembacaan per baris (sampai ditemukan 'carriage return') ke layar, sampai *end of file*.

Anda bisa membuat sendiri file teks-nya menggunakan notepad atau editor C++ lalu disimpan sebagai file teks dengan ekstensi .txt .

3. Operasi Baca Tulis (I/O) ke File Biner bertipe Integer.

Note : Khusus mengenai fungsi *fopen()*, *putw()*, dan *getw()*.

Contoh program dengan TCWin 4.5 terlampir untuk membaca dari dan menulis ke file biner untuk menyimpan data bertipe integer adalah : file **putw_wi.cpp** (mengisi /

membuat file biner bertipe integer) dan file **getw_wi.cpp** (membaca isi file biner yang telah dibuat)).

Pada program **putw_wi.cpp**, bila file data yang dibuka belum ada maka akan dibuat oleh program.

(Jalankan program putw_wi.cpp terlebih dahulu sebelum Anda menjalankan program getw_wi.cpp !)

4. Operasi Baca Tulis (I/O) ke File Biner bertipe Float dan Record.

Fungsi untuk membaca dan menyimpan data **bertipe float** atau data **bertipe record** (struct(struktur) pada Turbo C++) ke file adalah *fread()* dan *fwrite()*.

Operasi baca tulis lain terhadap file **bertipe record** adalah : *fscanf()* dan *fprintf()*.

Bentuk deklarasi *fread()* dan *fwrite()* :

`int fread(void *buffer, int n, FILE *ptr_file);`

digunakan untuk membaca isi file bertipe float atau record.

fread membaca sejumlah *n* item data, masing-masing dalam ukuran byte untuk panjang dari input nama berupa variabel bertipe file yang diberikan ke dalam sebuah blok yang ditunjuk oleh *ptr*. Jumlah byte yang dibaca adalah (*n* * ukuran).

`int fwrite(void *buffer, int jum_byte, int n, FILE *ptr_file);`

digunakan untuk menyimpan data ke file bertipe float atau record.

dimana

- *buffer* adalah :
 - *pointer* yang menunjuk ke daerah memori yang akan ditempati data dari file disk (untuk *fread()*), atau
 - *pointer* yang menunjuk ke daerah memori yang berisi data yang akan disimpan ke file disk (*fwrite()*).
- *jum_byte* : menyatakan jumlah byte yang akan dibaca atau disimpan.
- *n* : menentukan banyaknya blok data berukuran *jum_byte* yang akan ditulis atau dibaca.
- *ptr_file* berupa pointer-ke-file yang berisi nilai keluaran dari *fopen()*.

Contoh program dengan TCWin 4.5 terlampir, yaitu : **fwrite_w.cpp**, bertujuan untuk menyimpan dan membaca data bertipe record ke / dari sebuah file bernama DAFBUKU.DAT.

Program ini telah dimodifikasi, menyesuaikan perbedaan penggunaan beberapa fungsi antara Turbo C++ 3.0 dengan TCWin 4.5.

Perhatikan parameter yang digunakan pada fungsi *fopen* yaitu "wb".

Hal tersebut berarti untuk data bertipe float atau record, disimpan pada file dengan mode biner.

Bentuk deklarasi *fscanf()* dan *fprintf()* :

```
int fscanf(FILE *stream, const char *format[, address, ...]);
```

digunakan untuk membaca isi file bertipe record. Mirip dengan penggunaan fungsi *scanf* yang biasa digunakan, perbedaannya adalah pada *fscanf* dicantumkan nama file sebagai sumber data yang akan dibaca.

```
int fprintf(FILE *stream, const char *format[, argument, ...]);
```

digunakan untuk menyimpan data ke file bertipe record. Mirip dengan penggunaan fungsi *printf* yang biasa digunakan, perbedaannya adalah pada *fprintf* dicantumkan namafile sebagai tujuan penulisan/pengisian data.

dimana

- *stream* adalah : nama variabel bertipe FILE yang digunakan.

Contoh program dengan TCWin 4.5 terlampir, yaitu : **filedat.cpp**, bertujuan untuk menyimpan dan membaca data bertipe record ke / dari sebuah file bernama MHS_JTK.DAT menggunakan fungsi *fprintf* dan *fscanf*.

Perhatikan parameter yang digunakan pada fungsi *fopen* yaitu "ab+".

Carilah pemahaman mengenai arti dari parameter "ab+".

Pelajari fungsi lainnya terkait operasi terhadap file, misalnya fungsi *fseek()* dll. !

CONTOH – CONTOH PROGRAM

```
/*-----*/
/* File Program : FGETS.CPP */
/* Contoh pemakaian fgets() untuk membaca isi file teks */
/* Author : SN */
/*-----*/

#include <stdio.h>
#include <conio.h>
#include <string.h>
#include <stdlib.h>

#define PANJANG 256

int main(void)
{
    FILE *f_teks;
    char string[PANJANG];
    char namafile[65];

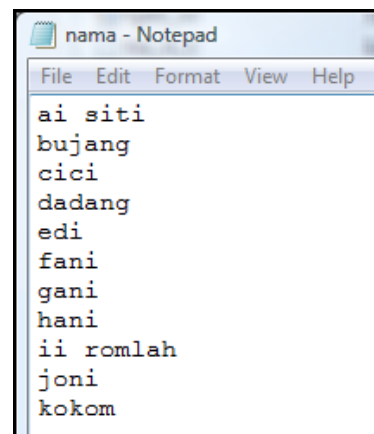
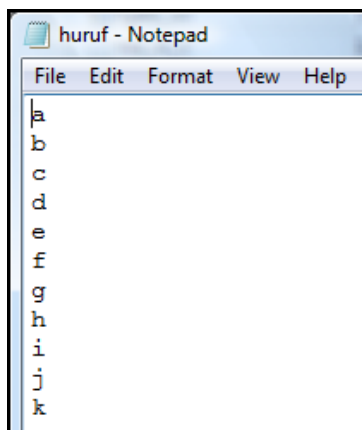
    printf("PROGRAM UNTUK MELIHAT ISI FILE TEKS\r\n");
    printf("Nama file : ");
    gets(namafile);
    /* pada output, tuliskan sebuah nama file yang akan dibaca, lengkap
    dengan ekstensinya! */

    /* buka file untuk dibaca */
    if ((f_teks=fopen(namafile, "rt")) == NULL)
    {
        printf("File tak dapat dibuka!\r\n");
        exit(1);
    }

    /* Baca string dari file selama masih ada */
    while ((fgets(string, PANJANG, f_teks)) != NULL)
    /* Tampilkan string ke layar */
    printf("%s\r", string);

    fclose(f_teks);      /* tutup file setiap selesai proses thd file */
    return 0;
}
```

Contoh file teks yang dibaca : huruf.txt dan nama.txt



```
/*-----*/
```

```

/* File Program : PUTW_WI.CPP                                     */
/* Contoh program untuk menyimpan data bertipe integer          */
/* ke file bertipe biner, menggunakan fungsi putw()             */
/* Author : SN                                                    */
/*-----*/

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <ctype.h>

int main(void)
{
    FILE *pf;
    int nilai;
    char jawab;

    clrscr();
    /* Jika file bilangan.dat tidak ada, maka ciptakan file biner nya*/
    /* Parameter wb menunjukkan bila file belum ada, maka akan dibuat*/
    if ((pf=fopen("BILANGAN.DAT", "wb")) == NULL)
    {
        printf("File gagal dibuat!\n");
        exit(1);
    }
    printf("Program untuk menyimpan data integer ke file bertipe");

    do {
        printf("\r\nBilangan yang akan disimpan : ");
        scanf("%d", &nilai); /* Baca nilaidari keyboard */
        putw(nilai, pf);      /* Tulis blangan ke file */
        printf("Memasukkan data lagi (Y/T)? ");
        jawab = toupper(getche()); /* Baca jawaban dari keyboard */
    }
    while (jawab == 'Y');
    fclose(pf);

    printf("\r\nData sudah disimpan pada file.\r\n");
    return 0;
}

/*-----*/
/* File Program : GETW_WI.CPP                                     */
/* Contoh program untuk membaca file biner berisi data integer   */
/* yang dibuat pada program PUTW_WI.CPP, menggunakan fungsi getw()*/
/* Author : SN                                                    */
/*-----*/

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>

int main(void)
{
    FILE *pf;
    int nilai;
    int nomor = 0;

    clrscr();
    /* Buka file biner 'bilangan.dat', untuk dibaca */
    /* Parameter rb untuk membaca file bertipe biner */
    if ((pf=fopen("BILANGAN.DAT", "rb")) == NULL)

```

```

    {
        printf("File gagal dibuka!\n");
        exit(1);
    }
    printf("Isi file BILANGAN.DAT :\r\n");
    while (1)
    {
        /* Baca sebuah nilai integer dari file BILANGAN.DAT */
        nilai = getw(pf);
        /* Jika akhir file, keluar dari loop */
        if (feof(pf) != NULL) break;
        /* Tampilkan ke layar */
        printf("%2d. %d\r\n", ++nomor, nilai);
    }

    fclose(pf); /* Tutup file */

    return 0;
}

/*-----*/
/* File Program : FWRITE_wi.CPP */
/* Contoh pemakaian fwrite() untuk menyimpan data bertipe record */
/* dan fread() untuk membaca isi file biner bertipe record */
/* Author : SN */
/*-----*/

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <ctype.h>
#include <string.h>

#define JUM_BLOK 1 /* sebuah record */

int main(void)
{
    FILE *f_dtbuku;
    char jawab;

    struct data_buku{
        char judul[26];
        char pengarang[30];
        int jumlah;
    }buku; /* variabel buku bertipe struktur */
    /* Buka file. Bila tidak ada, dibuat oleh TCWin 4.5 */
    if ( (f_dtbuku=fopen("DAF_BUKU.DAT","wb")) == NULL )
    {
        printf("File tidak dapat dibuat!\r\n");
        exit(1);
    }

    do {
        clrscr();
        printf("Judul buku      : "); fflush(stdin);
        gets(buku.judul);
        printf("Nama pengarang : "); fflush(stdin);
        gets(buku.pengarang);
        printf("Jumlah          : "); fflush(stdin);
        scanf("%d",&buku.jumlah);

        /* Rekam sebuah data bertipe record menggunakan fungsi fwrite*/

```

```

        fwrite(&buku, sizeof(buku), 1, f_dtbuku);

        printf("\r\nMemasukkan data lagi (Y/T)? ");fflush(stdin);
        jawab = toupper(getche());      /* Baca jawaban dari keyboard */
    }
    while (jawab == 'Y');
    fclose(f_dtbuku);    /* Tutup file */

    printf("\r\nData sudah disimpan pada file.\r\n");
    printf("\r\nLihat isi file (Y/T)? ");fflush(stdin);
    jawab = toupper(getche());      /* Baca jawaban dari keyboard */
    if (jawab == 'Y')
    { /* Buka file untuk dibaca isinya */
        if ( (f_dtbuku=fopen("DAF_BUKU.DAT","rb")) == NULL )
        {
            printf("File tidak dapat dibuka!\r\n");
            exit(1);
        }
        printf("\n");
        /* Ambil isi file menggunakan fungsi fread(),
           lalu tampilkan ke layar */
        while ( (fread(&buku, sizeof(buku), JUM_BLOK, f_dtbuku)) == JUM_BLOK )
            printf("%-26s %-20s %2d\r\n", buku.judul, buku.pengarang,
buku.jumlah);

        fclose(f_dtbuku); /* Tutup file */
    }

    return 0;
}

```

```

/* ----- */
/* Nama file      : FileDat.cpp                      */
/* Deskripsi      : Pengelolaan file data dalam C,    */
/*                meliputi entry, tampil dan sort    */
/* Author        : AD, SN                            */
/* Note          : Tampilan belum rapi, silakan di modifikasi! */
/* ----- */

```

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <conio.h>

```

```

typedef struct {char Kelas[3];
                char NIM[11];
                char JK;
                float IP; } Mhs;

```

```

void Entry_file();
void Sort_file();
void Tampil_file();

```

```

int main()
{
    int pil = 0;
    for (;;)
    {
        printf("\t\tMenu : \n");
        printf("\t 1. Entry data \n");
        printf("\t 2. Sort data \n");
    }
}

```



```

        printf("\t 3. Tampil isi file \n");
        printf("\t 4. S e l e s a i \n");
        printf("\n\t Pilihan : "); scanf("%d", &pil);
        switch (pil)
        {
            case 1 : { Entry_file();
                        break; }
            case 2 : { Sort_file();
                        break; }
            case 3 : { Tampil_file();
                        break; }
            default : { return(0);}
        }
    }
}

void Entry_file()
// Procedure untuk mengentrykan data ke file Mhs_jtk.dat
{
    Mhs dt;
    char lagi;
    FILE *f_mhs;

    if ((f_mhs=fopen("Mhs_jtk.dat", "ab+"))==NULL)
    {
        printf ("File tidak dapat dibuka\n");
        exit(1);
    }
    for (;;)
    {
        fflush(stdin);
        printf("Kelas : "); scanf("%s",&dt.Kelas);
        fflush(stdin);
        printf("N I M : "); scanf("%s",&dt.NIM);
        fflush(stdin);
        printf("Jenis Kelamin : "); scanf("%c", &dt.JK);
        printf("I P : "); scanf("%f", &dt.IP);
        // Sebelum nyimpan ke file, Hapus dulu isi penampung keyboard
        fflush(stdin);
        // Save to file
        fprintf(f_mhs, " %s %s %c %3.2f", dt.Kelas, dt.NIM, dt.JK,dt.IP);

        printf("Entry data lagi ? (Y/T) ");
        lagi = getche();
        printf("\n");
        if (lagi == 't' || lagi == 'T')
            break;
    }
    fclose(f_mhs);
}

void Tampil_file()
// Procedure untuk Menampilkan isi file mahasiswa
{
    Mhs dt;
    FILE *f_mhs;

    if ((f_mhs=fopen("Mhs_jtk.dat", "rb"))==NULL)
    {
        printf ("File tidak dapat dibuka\n");
        exit(1);
    }
}

```

```

    }
    printf ("\nISI FILE MHS_JTK.dat ADALAH : \n\n");
    printf ("Kelas      NIM      JK      IP \n");

    while (!feof(f_mhs))
    {
        fscanf(f_mhs,"%s %s %c %f", &dt.Kelas, &dt.NIM, &dt.JK, &dt.IP );
        printf("%s      %s      %c      %3.2f\n", dt.Kelas, dt.NIM, dt.JK,dt.IP);
    }
    fclose(f_mhs);
}

void Sort_file()
//sort file berdasarkan Kelas+NIM (Asc)
{
    FILE *f_mhs;
    long i=0, j;
    Mhs dt, Temp;

    if ((f_mhs=fopen("Mhs_jtk.dat", "rb+"))==NULL)
    {
        printf ("File tidak dapat dibuka\n");
        exit(1);
    }
    while (!feof(f_mhs))
    {
        fseek(f_mhs, (i*(sizeof(Mhs)+1))+i, SEEK_SET);
        fscanf(f_mhs,"%s %s %c %f", &dt.Kelas, &dt.NIM, &dt.JK, &dt.IP );
        j = i;
        while (!feof(f_mhs))
        {
            j++;
            fseek(f_mhs, (j*(sizeof(Mhs)+1))+j, SEEK_SET);
            fscanf(f_mhs,"%s %s %c %f",&Temp.Kelas,&Temp.NIM,&Temp.JK,&Temp.IP);
            if ((strcmp(dt.Kelas,Temp.Kelas)>0) ||
                ((strcmp(dt.Kelas,Temp.Kelas)==0) && (strcmp(dt.NIM,Temp.NIM)>0)))
            {
                fseek(f_mhs, i*(sizeof(Mhs)+1)+i, SEEK_SET);
                fprintf(f_mhs," %s %s %c %3.2f",Temp.Kelas,Temp.NIM,Temp.JK,Temp.IP);
                fseek(f_mhs, j*(sizeof(Mhs)+1)+j, SEEK_SET);
                fprintf(f_mhs, " %s %s %c %3.2f", dt.Kelas, dt.NIM, dt.JK,dt.IP);
                strcpy(dt.Kelas,Temp.Kelas);
                strcpy(dt.NIM,Temp.NIM);
                dt.JK = Temp.JK;
                dt.IP = Temp.IP;
            }
        }
        if (i < j)
        {
            fseek(f_mhs, 0, SEEK_SET);
            i++;
        }
    }
    printf("Sort File Berdasarkan Kelas dan NIM sudah selesai, hasilnya : ");
    fclose(f_mhs);
    Tampil_file();
}

```