

CASE STUDY I – BOOK SALES

LAPORAN STUDI KASUS MATA KULIAH COMP6360 – ALGORITHM AND
PROGRAMMING KELAS BE20



Oleh :

2440070125 – MARCEL WEISANG

2440040274 – RICHARD TANDY JAPUTRA

2440056902 – INTAN SALIYA UTOMO

Semester Ganjil 2020/2021

MALANG

Daftar Isi

GAMBARAN UMUM	1
1.1. Latar Belakang	1
1.2. Tujuan	1
1.3. Dasar Teori.....	1
1.4. Perancangan Program	2
1.4.1. Outline Solusi.....	2
1.4.2. Pengembangan Algoritma.....	3
1.4.3. Penjelasan Fitur program	4
1.5. Pembahasan Program.....	4
HASIL.....	9
2.1. Dokumentasi Hasil Program	9
2.2. Source Code Program	10
PEMBAGIAN TUGAS ANGGOTA.....	20

GAMBARAN UMUM

1.1. Latar Belakang

Dalam menjalankan suatu perusahaan, manajemen yang baik sangat diperlukan untuk merekam perkembangan dan penyusutan strategi usaha yang digunakan. Melihat pesatnya perkembangan teknologi saat ini, banyak sekali jalan yang dapat dipilih untuk mempermudah urusan yang sedang dihadapi. Teknologi informasi membuka seluruh jalan yang diinginkan untuk menunjang segala strategi dengan detail peluang yang cukup tinggi.

Perekaman serta pengarsipan data harus cukup diperhatikan, terutama apabila perusahaan aktif melakukan pengolahan mobilitas data. Dalam bidang penjualan buku dibutuhkan data dari setiap waktu, nama, dan data transaksi lain yang harus direkam. Frekuensi keluar masuk data cukup tinggi terjadi tiap periodenya. Maka dibutuhkan suatu metode yang efektif untuk memaksimalkan perintisan usaha penjualan buku ini.

Dengan pendekatan teknologi informasi, proses perekaman data penjualan buku akan dibantu oleh suatu program yang akan memudahkan perusahaan untuk melakukan penghitungan transaksi serta pendataan konsumen tiap harinya.

1.2. Tujuan

Adapun tujuan dari laporan ini adalah :

1. Dapat menganalisa suatu permasalahan dan merancang suatu penyelesaian untuk masalah terkait.
2. Mengaplikasikan algoritma penyelesaian masalah menjadi suatu aplikasi pemrograman bahasa C sederhana.
3. Dapat menyusun suatu pemrograman bahasa C sederhana dengan sintaksis yang sesuai serta fungsi yang tepat.

1.3. Dasar Teori

Algoritma merupakan suatu kumpulan atau rangkaian langkah – langkah logis yang digunakan untuk menyelesaikan suatu permasalahan yang melibatkan matematika dan logika. Umumnya algoritma diawali dengan suatu input data serta deskripsinya kemudian akan dilanjutkan dengan proses kalkulasi yang telah diatur dan hasil akhirnya akan dicetak sesuai dengan format yang dibuat.

Pemrograman merupakan suatu proses dalam menyusun, menguji atau memodifikasi suatu kode dalam pada suatu mesin yang menghasilkan suatu perintah untuk dijalankan. Pemrograman sendiri bertujuan untuk menjalankan perintah dari user. Hal tersebut dapat berupa penyelesaian suatu masalah, penyimpanan data, atau hiburan. Perintah yang diberikan pada mesin harus logis dan matematis agar algoritma yang dihasilkan dapat dijalankan tanpa adanya error.

Bahasa C merupakan salah satu bahasa pemrograman yang paling fleksibel untuk membuat berbagai aplikasi. Mampu untuk dijalankan di hampir semua komputer membuat penggunaan bahasa C sangat umum digunakan oleh banyak *programmer*. Berbagai macam tujuan juga dapat dilakukan menggunakan bahasa C. Mulai dari aplikasi sederhana, rangkaian system, kecerdasan buatan (*artificial intelligence*), database, game, bahkan virus dapat dibuat hanya dengan menggunakan bahasa C. Syntax yang tergolong sederhana membuat bahasa C mudah untuk diadaptasi menjadi bahasa pemrograman lainnya.

1.4. Perancangan Program

1.4.1. Outline Solusi

Dari masalah yang telah dianalisa, dibutuhkan suatu program untuk melakukan pendataan dari penjualah buku serta mencetak laporannya. Kemudian penjualan tersebut akan dicatat pada sebuah file yang bernama *sales.txt*. Adapun langkah – langkah yang dibutuhkan untuk menyusun program ini adalah sebagai berikut :

- Start.
- Menampilkan menu aktivitas yang bisa dilakukan.
- Memilih aktivitas yang akan dilakukan.
- Menampilkan halaman menu 1
- Memasukkan pilihan buku.
- Menghitung total harga pesanan.
- Memasukkan tanggal transaksi
- Memasukkan nama pelanggan.
- Menghitung jumlah pesanan.
- Mengirim data pada file *sales.txt*.
- Menampilkan halaman menu 2
- Mengurutkan data penjualan buku berdasarkan tanggal transaksi.
- Menampilkan data penjualan buku berdasarkan tanggal transaksi pada file *sales.txt*.
- Menampilkan halaman menu 3
- Mengurutkan data penjualan buku berdasarkan penjualan paling banyak.

- Menampilkan data penjualan buku berdasarkan penjualan paling banyak pada file sales.txt.
- End.

1.4.2. Pengembangan Algoritma

a. Tampilan menu utama

Pada awal program, pengguna akan disajikan ucapan selamat datang serta menu utama dari program. Pada bagian menu utama pengguna dapat melihat fitur apa saja yang tersedia. Fitur yang akan ditampilkan adalah menu untuk pembelian buku dan pencetakan laporan penjualan baik berdasarkan tanggal transaksi maupun berdasarkan jumlah penjualan produk terbanyak.

Untuk memilih fitur yang tersedia, pengguna akan diminta untuk menuliskan kode yang mewakili menu terpilih dan pengguna akan segera diarahkan pada halaman terkait. Jika pengguna memasukkan kode selain menu, maka program akan otomatis memberikan peringatan agar pengguna memasukkan kode baru yang tersedia.

b. Tampilan halaman menu 1

Saat pengguna tiba di halaman menu 1, yaitu menu pembelian buku, pengguna akan diberikan pilihan buku yang tersedia untuk dipesan. Pengguna juga akan dapat melihat keterangan jenis buku serta harganya. Untuk melakukan pemesanan, pengguna hanya perlu memasukkan kode buku yang ingin dipesan serta jumlah bukunya. Kemudian pengguna akan diberikan pilihan untuk melakukan pembelian buku lain atau tidak. Jika pengguna ingin melakukan pembelian kembali, maka pengguna akan diminta untuk memasukkan data buku yang ingin dibeli kembali. Jika tidak, pengguna akan diminta memasukkan tanggal transaksi dilakukan dengan format yang tersedia serta nama pelanggan. Disini pembeli akan kembali diberi pilihan untuk kembali ke menu utama atau tidak. Jika memilih untuk kembali ke menu utama maka pengguna akan mengulang proses seperti sebelumnya. Jika tidak, maka program akan berakhir dan akan otomatis tertutup.

c. Tampilan halaman menu 2

Pada menu ini pengguna akan melihat data penjualan yang telah dilakukan. Pengguna dapat melihat tabel berisi tanggal transaksi, nama pelanggan, jenis buku, nama buku, jumlah buku, harga buku, dan total transaksi yang telah berurutan

berdasarkan tanggal transaksi. Setelah puas melihat data yang tertera, pengguna akan diberikan pilihan untuk kembali ke menu utama atau mengakhiri program.

d. Tampilan halaman menu 3

Pada menu ini pengguna juga akan melihat tabel pembelian buku yang sama seperti pada menu 2, tetapi telah diurutkan berdasarkan jumlah pembelian buku terbanyak. Urutan data akan menampilkan data pembelian buku dari yang paling banyak hingga paling sedikit.

1.4.3. Penjelasan Fitur program

Berikut merupakan fitur yang terdapat pada program :

- **Pembelian buku**
Pengguna dapat melakukan penambahan data untuk penjualan buku.
- **Pencetakan laporan pembelian berdasarkan tanggal transaksi**
Pada fitur ini, data yang telah terekam akan diurutkan berdasarkan tanggal transaksi.
- **Pencetakan laporan pembelian berdasarkan penjualan terbanyak**
Fitur ini juga akan mengurutkan data penjualan buku berdasarkan hasil penjualan buku terbanyak.

1.5. Pembahasan Program

a. Pemilihan library untuk program

Standart library yang akan digunakan pada program ini yaitu :

stdio.h untuk melakukan input serta menampilkan output dari program.

stdlib.h untuk mengolah data bertipe *string* menjadi bentuk *integer*.

string.h untuk melakukan perintah pada data bertipe *string*.

time.h untuk mendefinisikan tanggal waktu penjalanan program.

b. Fungsi untuk menampilkan menu utama

Pada fungsi ini akan ditampilkan ucapan selamat datang serta menu untuk aktivitas yang dapat dilakukan.

Disini input menu akan dibaca sebagai *integer* yang tertera pada menu untuk mewakili menu pilihan pengguna.

c. Fungsi untuk verifikasi menu utama

Fungsi ini dibuat untuk membatasi input data pengguna agar sesuai dengan menu yang tersedia. Jika input yang dimasukkan pengguna tidak valid maka error akan ditampilkan. Pilihan menu dan validasi kondisi akan dideklarasikan sebagai *integer*.

Setelah pilihan menu dan validasi kondisi dideklarasikan, akan dibuat pembatasan input menggunakan *looping* pada setiap menu yang tersedia. Fungsi yang akan dibuat menggunakan parameter bertipe *integer*.

Lalu, pengguna memasukkan input selain 1, 2, atau 3 maka pengguna akan diberikan peringatan dan akan diarahkan untuk memasukkan input baru. Kondisi dan pilihan menu pada fungsi diatas menjadi bentuk pointer untuk dilakukan verifikasi kondisi pada fungsi selanjutnya. Verifikasi akan dilakukan menggunakan proses *selection*.

Setelah melewati semua fungsi yang dibuat, pengguna akan segera diarahkan pada halaman menu pilihan. Jika tidak, maka peringatan error akan muncul dan pengguna harus mengulang proses dari awal kembali.

Untuk menutup program, akan ditampilkan tanda akhir dari program maka nantinya pengguna akan tahu bahwa program telah benar - benar berakhir.

d. Fungsi untuk menu pendataan penjualan buku

Pada fungsi ini pengguna akan melakukan serangkaian proses untuk menambahkan data pembelian buku. Fungsi akan diawali dengan deklarasi variabel yang akan digunakan pada fungsi ini. kode buku, jumlah buku, harga buku, dan total pembayaran akan dideklarasikan sebagai *integer* sedangkan judul buku, jenis buku, nama buku dan nama pelanggan akan dideklarasikan sebagai *string*.

Disini pengguna akan melihat daftar buku yang tersedia beserta deskripsinya dan akan dipersilahkan untuk memilih buku yang diinginkan dengan menginput kode buku terkait.

Agar kode buku dapat mewakili nama buku yang tertera serta memuat nilai dari harga setiap buku, maka akan dibuat *switch case* yang memuat deskripsi setiap buku. Jika kode buku yang dimasukkan tidak tertera, maka pengguna akan mendapat peringatan adanya kesalahan dalam proses input kode dan akan diberikan pilihan untuk kembali ke menu utama atau mengakhiri program. Jika kode yang diinput valid, maka pengguna akan diminta untuk memasukkan jumlah buku yang ingin dibeli.

Pengguna akan mendapatkan pilihan untuk membeli buku lagi atau tidak. Agar program dapat mendapatkan nilai baru, maka nilai variabel sebelumnya akan disimpan kedalam *array*. Disini total pembayaran juga akan dihitung dengan melakukan pengalian kepada jumlah buku dan harga buku. Total pembayaran yang telah dihitung juga akan disimpan kedalam *array*.

Pilihan yang diinput oleh pengguna akan divalidasi dengan menggunakan *selection*. Jika pengguna memilih untuk membeli buku lagi maka fungsi ini akan ditutup dan pengguna akan diperbolehkan memilih buku lain untuk dibeli. Jikapengguna memilih untuk tidak memilih buku lain, maka pengguna akan diminta untuk memasukkan tanggal transaksi sesuai format yang ditunjukkan. Program juga akan secara otomatis menampilkan tanggal setempat sebagai informasi tambahan. Setelah memasukkan tanggal pengguna akan diminta untuk memasukkan nama pembeli.

Disini, program akan menambahkan data yang telah dimasukkan untuk diperbarui dalam laporan. Oleh karena itu program akan mulai membuka file *sales.txt* yang sudah ada dan juga membuat file baru dengan nama *sales_temp.txt* untuk melakukan penambahan data. Program akan membaca data pada file *sales.txt* dan juga membuat alokasi memori sesuai jumlah data yang telah terbaca. Dua garis awal yang berisi jumlah data dan kepala tabel akan diabaikan dengan menggunakan *buffer*.

Semua data lama akan diambil dengan menggunakan *looping* dan mencetaknya pada file *sales_temp.txt*. Disini pengguna akan melihat tabel yang serupa seperti yang terdapat dalam file *sales.txt* namun dengan penambahan data yang baru saja diinput. Setelah data lama dan data baru dicetak di file *sales_temp.txt*, looping akan berhenti dilakukan. Untuk mengakhiri proses pencetakan laporan, file *sales.txt* dan file *sales_temp.txt* akan ditutup.

Karena semua data terbaru telah dipindahkan pada file *sales_temp.txt*, file *sales.txt* tidak lagi digunakan. Maka disini file *sales.txt* akan dihapus. Agar file *sales_temp.txt* dapat dibaca oleh program pada proses pendataan selanjutnya, maka file *sales_temp.txt* akan diubah menjadi *sales.txt* seperti nama file awal.

Untuk mengakhiri semua proses diatas, program akan memberikan keterangan bahwa data yang dimasukkan oleh pengguna berhasil ditambahkan kedalam laporan. Pengguna akan diberi pilihan untuk kembali ke menu utama atau mengakhiri program.

e. Fungsi untuk pencetakan laporan penjualan buku

Pada fungsi ini data yang telah diinput oleh pengguna akan dimasukkan dalam file bernama *sales.txt*. Fungsi ini menggunakan parameter *integer*.

Variabel *b* pada fungsi mewakili jenis sorting yang akan dilakukan pengguna. Setelah fungsi dibuka, semua variable akan dideklarasikan. Data buku akan dideklarasikan sebagai *string* dan jumlah data dideklarasikan sebagai *integer*.

Setelah proses deklarasi selesai, file *sales.txt* akan dibuka dan program akan diatur untuk membaca data yang tersedia. Disini program akan mulai membaca data yang yang diinput pengguna. Jumlah data yang diinginkan pengguna akan dibaca dan akan diberikan alokasi memori untuk data rincian penjualan nantinya.

Setelah alokasi memori data diberikan maka tabel data penjualan buku akan mulai dibuat dan melakukan *looping* hingga data memenuhi memori yang tersedia. Untuk memudahkan proses sorting nantinya, maka data di line pertama dan kedua yang berisi jumlah data dan kepala tabel akan diabaikan. Setelah memenuhi alokasi memori yang ada, proses pencetakan laporan akan selesai. File *sales.txt* akan ditutup.

f. Fungsi untuk sorting data berdasarkan tanggal transaksi

Pada fungsi ini data buku yang ada akan diurutkan berdasarkan tanggal transaksi dan jumlah penjualan terbanyak sesuai dengan menu yang telah ditampilkan sebelumnya. Jumlah setiap jenis buku akan dibandingkan menggunakan *selection* dan akan diurutkan dari jumlah terbanyak hingga terkecil (*descending*) hingga data terakhir menggunakan *looping*. Jenis sorting akan dilakukan berdasarkan input yang dimasukkan oleh pengguna. Disini *selection* akan dilakukan untuk mencocokkan menu pilihan pengguna dengan sorting yang akan dilakukan.

- Sorting berdasarkan tanggal transaksi

Pada proses *sorting*, tahun disorting secara *descending*. Jika tahun transaksi sama, maka akan disorting berdasarkan bulan secara *descending*. Dan jika bulan pembelian sama maka akan disorting berdasarkan hari pembelian. Dengan begitu data akan menjadi urut dari transaksi paling baru hingga paling lama.

- Sorting berdasarkan jumlah pembelian terbanyak

Pada proses *sorting* diatas, jumlah pembelian terbanyak disorting secara *descending*. Setelah proses *sorting* berakhir maka data akan urut dari jenis buku dengan pembelian terbanyak hingga jenis buku dengan jumlah pembelian paling sedikit.

Jenis *sorting* akan dilakukan berdasarkan *input* yang dimasukkan oleh pengguna. Disini *selection* akan dilakukan untuk mencocokkan menu pilihan pengguna dengan *sorting* yang akan dilakukan.

Disini, jika pengguna menginput menu 3 maka *sorting* akan dilakukan berdasarkan jumlah pembelian dan jika pengguna memilih menu 2 maka *sorting* akan dilakukan berdasarkan tanggal transaksi.

HASIL

2.1. Dokumentasi Hasil Program

```
C:\Users\HP\Downloads\Gas(2) (extract.me)\Gas\library.exe
*---- Selamat Datang Di Aplikasi Penjualan Buku ----*
Mau Ngapain:
[1] Menjual Buku
[2] Cetak Laporan Buku Berdasarkan Tanggal Transaksi
[3] Cetak Laporan Buku Berdasarkan Penjualan Terbanyak
```

Tampilan menu utama program

```
C:\Users\HP\Downloads\Gas(2) (extract.me)\Gas\library.exe
Silahkan pilih buku yang mau dibeli: (Buku Ke-1)

No    Nama Buku      Jenis Buku  Harga Buku
1     Journalism     NoFiksi    10000
2     Brontosaurus   NoFiksi    20000
3     Soekarno       NoFiksi    72500
4     Keberangkatan  Fiksi      100000
5     Noblesse       Fiksi      5000
6     Endessor       Fiksi      25000

Silahkan Pilih Kode buku yang ingin dibeli: _
```

Tampilan awal menu 1 (menu pembelian buku).

```
C:\Users\HP\Downloads\Gas(2) (extract.me)\Gas\library.exe
Silahkan pilih buku yang mau dibeli: (Buku Ke-1)

No    Nama Buku      Jenis Buku  Harga Buku
1     Journalism     NoFiksi    10000
2     Brontosaurus   NoFiksi    20000
3     Soekarno       NoFiksi    72500
4     Keberangkatan  Fiksi      100000
5     Noblesse       Fiksi      5000
6     Endessor       Fiksi      25000

Silahkan Pilih Kode buku yang ingin dibeli: 1
Jumlah buku yang mau dibeli: 2
Mau Beli Buku Lagi(y/n) ? n
Masukan tanggal transaksi (format : 09-01-2021): 09-01-2021
Masukan nama pelanggan: Customer
```

Tampilan menu 1 satu ketika semua input telah dilakukan.

```
C:\Users\HP\Downloads\Gas(2) (extract.me)\Gas\library.exe
Data berhasil ditambahkan
Kembali ke menu utama(y/n) ?
```

Tampilan menu 1 ketika data berhasil ditambahkan kedalam laporan.

```
C:\Users\HP\Downloads\Gas(2) (extract.me)\Gas\library(1).exe
*----- Cetak Penjualan Buku -----*
No    Tanggal Transaksi  Nama Pelanggan  Jenis Buku  Nama Buku      Jumlah Buku  Harga Buku  Total Pembayaran
1     3-12-2020          intan           NoFiksi    Soekarno       3            72500      217500
2     25-3-2020          dia            NoFiksi    Brontosaurus   1            20000      20000
3     9-8-2018           saya           NoFiksi    Brontosaurus   2            20000      40000
4     9-8-2018           saya           Fiksi      Noblesse       1            5000       5000
5     9-7-2015           makoto         Fiksi      Endessor       2            25000      50000
6     19-4-2015          kita           Fiksi      Noblesse       1            5000       5000
7     20-1-2014          Rio            NoFiksi    Journalism     2            10000      20000
8     18-11-1982         weisang        Fiksi      Keberangkatan  4            100000     400000
Kembali ke menu utama(y/n) ?
```

Tampilan menu 2.

C:\Users\HP\Downloads\Gas(2) (extract.me)\Gas\library.exe

----- Cetak Penjualan Buku -----*

No	Tanggal Transaksi	Nama Pelanggan	Jenis Buku	Nama Buku	Jumlah Buku	Harga Buku	Total Pembayaran
1	18-11-1982	weisang	Fiksi	Keberangkatan	4	100000	400000
2	3-12-2020	intan	NoFiksi	Soekarno	3	72500	217500
3	20-1-2014	Rio	NoFiksi	Journalism	2	10000	20000
4	9-7-2015	makoto	Fiksi	Endessor	2	25000	50000
5	9-8-2018	saya	NoFiksi	Brontosaurus	2	20000	40000
6	9-1-2021	intan	NoFiksi	Brontosaurus	2	20000	40000
7	19-4-2015	kita	Fiksi	Noblesse	1	5000	5000
8	9-8-2018	saya	Fiksi	Noblesse	1	5000	5000
9	25-3-2020	dia	NoFiksi	Brontosaurus	1	20000	20000

Kembali ke menu utama(y/n) ?

Tampilan menu 3.

2.2. Source Code Program

- **header.h**

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include <time.h>
#include "fungsi.h"

// "os" = option selection,
int os, condition;
```

- **main.h**

```
#include "header.h"

int main(int argc, char *argv[]) {
    // Halaman Utama
    int stop_p = 0;
    while(stop_p == 0){
        welcome();
        option();
        printf("\n");

        // Pilih Opsi Menu
        scanf("%d", &os);

        // Verifikasi inputan user
        verif_opsi(os, &os, &condition);
        if(condition == 1){
            // Menuju halaman sesuai opsi yang telah dipilih
            pilih_opsi_menu(os, &stop_p);
        }else{
            printf("Err : Terjadi Kesalahan");
        }
    }
    system("cls");
    printf("akhir dari program...");
    return 0;
}
```

- **fungsi.h**

```
// Struct untuk Laporan buku
typedef struct{
```

```

    int hari;
    int bulan;
    int tahun;
    char *judul[20];
    char *nama_pelanggan[20];
    char *jenis_buku[20];
    char *nama_buku[20];
    int jumlah_buku;
    int harga_buku;
    int total_pembayaran;
} Buku;

//struct untuk pembelian buku

void welcome(){
    printf("-----\n");
    printf("*---- Selamat Datang Di Aplikasi Penjualan Buku ----*\n");
    printf("\n-----\n");
}

void option(){
    printf("Mau Ngapain:\n\n");
    //Ops Menu
    printf("[1] Menjual Buku\n[2] Cetak Laporan Buku Berdasarkan Tanggal Transaksi\n[3] Cetak Laporan Buku Berdasarkan Penjualan Terbanyak");
}

void verif_opsi(int a, int *os, int *condition){
    //looping dengan pesan error
    while(a>3 || a<1){
        system("cls");
        welcome();
        printf("Error: Tidak ada opsi dengan pilihan \"%d\"\n",a);
        option();
        printf("\n");
        scanf("%d", &a);
    }
    //kasih pointer value
    *os = a;
    *condition = 1;
}

void cetak_laporan_buku(int b){
    char buffer[100], tulisan[30];
    char tmp2[20];
    int i, j, tmp;
    int jumlah_data, a;

    system("cls");

    printf("-----\n");
    printf("-----\n");
    printf("*----- Cetak Penjualan Buku -----*\n\n");

    FILE *file = fopen("sales.txt", "r");

```

```

fscanf(file, "%s %d", &tulisan, &jumlah_data);
Buku *isi = malloc(sizeof(Buku) * jumlah_data);

//      skip 2 line pertama
fgets(buffer, 100, file);
fgets(buffer, 100, file);

printf("No\tTanggal Transaksi\tNama Pelanggan\t\tJenis Buku\tNama
Buku\t\tJumlah Buku\tHarga Buku\tTotal Pembayaran\n\n");

//mengambil data lama
for(a = 0; a<jumlah_data; a++){
    fscanf(file, "%d-%d-%d %s %s %s %d %d %d", &isi[a].hari,
&isi[a].bulan, &isi[a].tahun, &isi[a].nama_pelanggan,
&isi[a].jenis_buku, &isi[a].nama_buku, &isi[a].jumlah_buku,
&isi[a].harga_buku, &isi[a].total_pembayaran);
}

if(b == 3){
    //urutkan berdasarkan jumlah terbanyak
    for(i=0; i<jumlah_data; i++)
    {
        for(j=i+1; j<jumlah_data; j++)
        {
            if(isi[i].jumlah_buku < isi[j].jumlah_buku)
            {
                //jumlah buku
                tmp = isi[i].jumlah_buku;
                isi[i].jumlah_buku = isi[j].jumlah_buku;
                isi[j].jumlah_buku = tmp;

                //hari
                tmp = isi[i].hari;
                isi[i].hari = isi[j].hari;
                isi[j].hari = tmp;

                //bulan
                tmp = isi[i].bulan;
                isi[i].bulan = isi[j].bulan;
                isi[j].bulan = tmp;

                //tahun
                tmp = isi[i].tahun;
                isi[i].tahun = isi[j].tahun;
                isi[j].tahun = tmp;

                //nama pelanggan
                strcpy(tmp2, isi[i].nama_pelanggan);
                strcpy(isi[i].nama_pelanggan, isi[j].nama_pelanggan);
                strcpy(isi[j].nama_pelanggan, tmp2);

                //jenis buku
                strcpy(tmp2, isi[i].jenis_buku);
                strcpy(isi[i].jenis_buku, isi[j].jenis_buku);
                strcpy(isi[j].jenis_buku, tmp2);

                //nama buku
                strcpy(tmp2, isi[i].nama_buku);
                strcpy(isi[i].nama_buku, isi[j].nama_buku);
                strcpy(isi[j].nama_buku, tmp2);
            }
        }
    }
}

```



```

        //total pembayaran
        tmp = isi[i].total_pembayaran;
        isi[i].total_pembayaran = isi[j].total_pembayaran;
        isi[j].total_pembayaran = tmp;
    }
}

//bulan
for(i=0; i<jumlah_data; i++)
{
    for(j=i+1; j<jumlah_data; j++)
    {
        if(isi[i].tahun == isi[j].tahun &&
isi[i].bulan<isi[j].bulan)
        {
            //jumlah buku
            tmp = isi[i].jumlah_buku;
            isi[i].jumlah_buku = isi[j].jumlah_buku;
            isi[j].jumlah_buku = tmp;

            //hari
            tmp = isi[i].hari;
            isi[i].hari = isi[j].hari;
            isi[j].hari = tmp;

            //bulan
            tmp = isi[i].bulan;
            isi[i].bulan = isi[j].bulan;
            isi[j].bulan = tmp;

            //tahun
            tmp = isi[i].tahun;
            isi[i].tahun = isi[j].tahun;
            isi[j].tahun = tmp;

            //nama pelanggan
            strcpy(tmp2, isi[i].nama_pelanggan);
            strcpy(isi[i].nama_pelanggan, isi[j].nama_pelanggan);
            strcpy(isi[j].nama_pelanggan, tmp2);

            //jenis buku
            strcpy(tmp2, isi[i].jenis_buku);
            strcpy(isi[i].jenis_buku, isi[j].jenis_buku);
            strcpy(isi[j].jenis_buku, tmp2);

            //nama buku
            strcpy(tmp2, isi[i].nama_buku);
            strcpy(isi[i].nama_buku, isi[j].nama_buku);
            strcpy(isi[j].nama_buku, tmp2);

            //harga buku
            tmp = isi[i].harga_buku;
            isi[i].harga_buku = isi[j].harga_buku;
            isi[j].harga_buku = tmp;

            //total pembayaran
            tmp = isi[i].total_pembayaran;
            isi[i].total_pembayaran = isi[j].total_pembayaran;

```



```

isi[j].total_pembayaran = tmp;
    }
}

//hari
for(i=0; i<jumlah_data; i++)
{
    for(j=i+1; j<jumlah_data; j++)
    {
        if(isi[i].tahun == isi[j].tahun && isi[i].bulan ==
isi[j].bulan && isi[i].hari < isi[j].hari)
        {
            //jumlah buku
            tmp = isi[i].jumlah_buku;
            isi[i].jumlah_buku = isi[j].jumlah_buku;
            isi[j].jumlah_buku = tmp;

            //hari
            tmp = isi[i].hari;
            isi[i].hari = isi[j].hari;
            isi[j].hari = tmp;

            //bulan
            tmp = isi[i].bulan;
            isi[i].bulan = isi[j].bulan;
            isi[j].bulan = tmp;

            //tahun
            tmp = isi[i].tahun;
            isi[i].tahun = isi[j].tahun;
            isi[j].tahun = tmp;

            //nama pelanggan
            strcpy(tmp2, isi[i].nama_pelanggan);
            strcpy(isi[i].nama_pelanggan, isi[j].nama_pelanggan);
            strcpy(isi[j].nama_pelanggan, tmp2);

            //jenis buku
            strcpy(tmp2, isi[i].jenis_buku);
            strcpy(isi[i].jenis_buku, isi[j].jenis_buku);
            strcpy(isi[j].jenis_buku, tmp2);

            //nama buku
            strcpy(tmp2, isi[i].nama_buku);
            strcpy(isi[i].nama_buku, isi[j].nama_buku);
            strcpy(isi[j].nama_buku, tmp2);

            //harga buku
            tmp = isi[i].harga_buku;
            isi[i].harga_buku = isi[j].harga_buku;
            isi[j].harga_buku = tmp;

            //total pembayaran
            tmp = isi[i].total_pembayaran;
            isi[i].total_pembayaran = isi[j].total_pembayaran;
            isi[j].total_pembayaran = tmp;
        }
    }
}

```



```

        break;
    case 3:
        strcpy(nama_buku, "Soekarno");
        strcpy(jenis_buku, "NoFiksi");
        harga_buku = 72500;
        break;
    case 4:
        strcpy(nama_buku, "Keberangkatan");
        strcpy(jenis_buku, "Fiksi");
        harga_buku = 100000;
        break;
    case 5:
        strcpy(nama_buku, "Noblesse");
        strcpy(jenis_buku, "Fiksi");
        harga_buku = 5000;
        break;
    case 6:
        strcpy(nama_buku, "Endessor");
        strcpy(jenis_buku, "Fiksi");
        harga_buku = 25000;
        break;
    default :
        printf("Terjadi Kesalahan");
        return 1;
}

printf("Jumlah buku yang mau dibeli: ");
scanf("%d", &jumlah_buku);

// penyimoanan data ke arr
strcpy(judul[count], nama_buku);
arr_jumlah_buku[count] = jumlah_buku;
arr_harga_buku[count] = harga_buku;
total_pembayaran[count] = jumlah_buku * harga_buku;
strcpy(jenis_buku[count], jenis_buku);

//konfirmasi kelanjutan beli buku
printf("Mau Beli Buku Lagi(y/n) ? ");
scanf(" %c", &lanjut);

if(lanjut == 'n'){
    //tidak lanjut

    //input tanggal transaksi
    printf("Masukan tanggal transaksi (format : %02d-%02d-%d): ", tm.tm_mday, tm.tm_mon + 1, tm.tm_year + 1900);
    scanf("%d-%d-%d", &hari, &bulan, &tahun);

    //input nama pelanggan
    printf("Masukan nama pelanggan: ");
    scanf("%s", &nama_pelanggan);
    char buffer[100], tulisan[30];

    //file start
    FILE *file_lama = fopen("sales.txt", "r");
    FILE *file_baru = fopen("sales_temp.txt", "w");

    fscanf(file_lama, "%s %d", &tulisan, &jumlah_data_lama);
    Buku *isi = malloc(sizeof(Buku) * jumlah_data_lama);

    //skip 2 line pertama

```



```

switch(os){
    case 1:
        halaman_pembelian();
        break;
    case 2:
        cetak_laporan_buku(os);
        break;
    case 3:
        cetak_laporan_buku(os);
        break;
    default :
        printf("Terjadi Kesalahan");
        return 1;
        break;
}

printf("Kembali ke menu utama(y/n) ? ");
scanf(" %c", &la);
if(la == 'n'){
    *stop = 1;
}else{
    system("cls");
}
}

```

PEMBAGIAN TUGAS ANGGOTA

Pendataan :

Marcel Weisang : Penyusunan program pendataan penjualan buku.

Richard Tandy Japutra : Penyusunan program pencetakan laporan penjualan buku terurut berdasarkan transaksi.

Intan Saliya Utomo : Penyusunan program pencetakan laporan penjualan buku terurut berdasarkan penjualan terbanyak.

Untuk pembuatan laporan disusun secara kolektif dalam satu waktu.