#include <stdio.h>

#include <string.h>

int input,idb;

int index = 0;

struct library{

    char title[100], author[100];

    int year, id;

};

library book[100];

void merge(int l, int m, int r){

    int N1 = m - l + 1;

    int N2 = r - m;

    char tempTitle\_L[N1][100], tempTitle\_R[N2][100], tempAuthor\_L[N1][100], tempAuthor\_R[N2][100];

    int tempId\_L[N1], tempId\_R[N2], tempYear\_L[N1], tempYear\_R[N2];

    for(int i = 0; i < N1; i++){

        strcpy(tempTitle\_L[i], book[l + i].title);

        strcpy(tempAuthor\_L[i], book[l + i].author);

        tempId\_L[i] = book[l + i].id;

        tempYear\_L[i] = book[l + i].year;

    }

    for(int j = 0; j < N2; j++){

        strcpy(tempTitle\_R[j], book[m + 1 + j].title);

        strcpy(tempAuthor\_R[j], book[m + 1 + j].author);

        tempId\_R[j] = book[m + 1 + j].id;

        tempYear\_R[j] = book[m + 1 + j].year;

    }

    int i = 0;

    int j = 0;

    int k = l;

    while(i < N1 && j < N2){

        if(strcmp(tempTitle\_L[i], tempTitle\_R[j]) < 0){

            strcpy(book[k].title, tempTitle\_L[i]);

            strcpy(book[k].author, tempAuthor\_L[i]);

            book[k].id = tempId\_L[i];

            book[k].year = tempYear\_L[i];

            i++;

        } else {

            strcpy(book[k].title, tempTitle\_R[j]);

            strcpy(book[k].author, tempAuthor\_R[j]);

            book[k].id = tempId\_R[j];

            book[k].year = tempYear\_R[j];

            j++;

        }

        k++;

    }

    while(i < N1){

        strcpy(book[k].title, tempTitle\_L[i]);

        strcpy(book[k].author, tempAuthor\_L[i]);

        book[k].id = tempId\_L[i];

        book[k].year = tempYear\_L[i];

        i++;

        k++;

    }

    while(j < N2){

        strcpy(book[k].title, tempTitle\_R[j]);

        strcpy(book[k].author, tempAuthor\_R[j]);

        book[k].id = tempId\_R[j];

        book[k].year = tempYear\_R[j];

        j++;

        k++;

    }

}

void mergesort(int l, int r){

    if(l < r){

        int m = l + (r - l) / 2;

        mergesort(l, m);

        mergesort(m + 1, r);

        merge(l, m, r);

    }

}

void printmenu(){

    printf("============================================================\n");

    for(int i = 0; i < index; i++){

        printf("%-5d- %-20s- %-20s- %-5d\n", book[i].id, book[i].title, book[i].author, book[i].year);

    }

    printf("============================================================\n");

}

void update(){

    printf("Input Book Id : ");

    scanf("%d", &idb);

    for(int i = 0; i < index; i++){

        if(idb == book[i].id){

            printf("New Book ID : ");

            scanf("%d", &book[i].id);

            printf("Book Title : ");

            scanf(" %[^\n]", &book[i].title);

            printf("Book Author : ");

            scanf(" %[^\n]", &book[i].author);

            printf("Year Publish : ");

            scanf("%d", &book[i].year);

        }

    }

}

void create(){

    FILE \*create;

    create = fopen("Library.txt", "r");

    if(create == NULL){

        create = fopen("Library.txt", "w");

    } else {

        while(!feof(create)){

            fscanf(create, "%d#%[^#]#%[^#]#%d\n", &book[index].id, &book[index].title, &book[index].author, &book[index].year);

            index++;

        }

    }

    fclose(create);

}

void write(){

    FILE \*write;

    write = fopen("Library.txt", "w");

    for(int i = 0; i < index; i++){

        fprintf(write, "%d | %s | %s | %d\n", book[i].id, book[i].title, book[i].author, book[i].year);

    }

    fclose(write);

}

void erase(){

    printf("Input Book Id : ");

    scanf("%d", &idb);

    for(int i = 0; i < index; i++){

        if(idb == book[i].id){

            for(int j = i; j < index - 1; j++){

                book[j].id = book[j + 1].id;

                strcpy(book[j].title, book[j + 1].title);

                strcpy(book[j].author, book[j + 1].author);

                book[j].year = book[j + 1].year;

            }

            index--;

        }

    }

}

int getmenu(){

    printf("LIBRARY INFORMATION\n");

    printf("==============================\n");

    printf("1.Input book record\n");

    printf("2.Display book record (Sorted)\n");

    printf("3.Update book record\n");

    printf("4.Erase book record\n");

    printf("5.Save and Exit\n");

    printf("==============================\n");

    printf("Pilih [1-5]? :");

    scanf("%d", &input);

    while(input < 1 || input > 5){

        printf("Pilih [1-5]? : ");

        scanf("%d", &input);

    }

    return input;

}

void InputBook(){

    if(index < 100){

        printf("Book ID : ");

        scanf(" %d", &book[index].id);

        printf("Book Title : ");

        scanf(" %[^\n]", &book[index].title);

        printf("Book Author : ");

        scanf(" %[^\n]", &book[index].author);

        printf("Year Publish : ");

        scanf(" %d", &book[index].year);

        index++;

    } else {

        printf("List buku sudah penuh");

    }

}

int main(){

    while (input >=1 || input <=5){

        getmenu();

        if(input == 1){

            printf("\n");

            InputBook();

            getchar(); getchar();

            write();

        }

        else if(input == 2){

            printf("\n");

            mergesort(0, index - 1);

            printmenu();

            getchar(); getchar();

        }

        else if(input == 3){

            printf("\n");

            printmenu();

            update();

            getchar(); getchar();

            write();

        }

        else if(input == 4){

            printf("\n");

            erase();

            printmenu();

            getchar(); getchar();

        }

        else if(input == 5){

            printf("Thank you, bye !");

            getchar(); getchar();

            write();

            break;

        }

    }

    return 0;

}