

**LAPORAN FINAL PROJECT PRAKTIKUM STRUKTUR DATA**

**“Aplikasi sederhana system manaejemn perpustakaan”**



**Disusun Oleh:**

**Serly Eka Putri**

**BP/NIM : 2023/23343083**

**Hari/Tanggal : Jum'at/ 31-05-2024**

**Sesi/Jam : 202323430157/ 08:50 - 10:30 WIB**

**Dosen Pengampu :**

**Randi Proska Sandra, M.Sc.**

**NIP. 221048**

**FAKULTAS TEKNIK**

**DEPARTEMEN TEKNIK ELEKTRONIKA**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**UNIVERSITAS NEGERI PADANG**

**2024**

## **Aplikasi Sederhana Sistem manajemen perpustakaan**

### **A. Latar Belakang**

Latar belakang dari program ini adalah kebutuhan untuk mengelola koleksi buku dalam sebuah perpustakaan dengan cara yang efisien dan terorganisir. Program ini dirancang untuk membantu perpustakaan dalam mengelola inventaris buku mereka, memastikan buku-buku dapat dengan mudah ditambahkan, dipinjam, dikembalikan, dan dicari oleh pengguna.

Berikut ini adalah beberapa latar belakang lebih rinci yang mendasari pengembangan program ini :

1. **Manajemen Inventaris Buku :**  
Program ini membantu dalam mencatat dan menyimpan informasi tentang setiap buku, termasuk judul, penulis, dan status peminjamannya.
2. **Peminjam dan Pengembalian Buku :**  
Program ini menyediakan fitur untuk meminjam dan mengembalikan buku, memastikan bahwa status peminjaman buku selalu terbaru dan akurat.
3. **Pencarian Buku :**  
Program ini memungkinkan pencarian buku berdasarkan judul atau penulis, memudahkan pengguna untuk menemukan buku yang mereka cari.
4. **Organisasi dan Pengurutan Buku :**  
Program ini memungkinkan buku diurutkan berdasarkan judul, membuatnya lebih mudah untuk ditemukan dan diakses.
5. **Kemudahan Pengguna :**  
Program ini dirancang dengan antarmuka yang sederhana dan mudah digunakan, memungkinkan pengguna dari berbagai latar belakang untuk mengoperasikannya tanpa memerlukan pelatihan teknis yang mendalam.
6. **Efisiensi dan Otomatisasi :**  
Program ini mengurangi beban kerja staf perpustakaan dan meningkatkan efisiensi operasional.

Program ini bertujuan untuk meningkatkan efisiensi pengelolaan perpustakaan, menyediakan akses yang lebih baik kepada pengguna, dan memastikan bahwa koleksi buku dapat diakses dan dikelola dengan cara yang sistematis dan terorganisir.

## B. Source Code

```
// Created_by_Serly Eka Putri_23343083

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

#define MAX_JUDUL_LENGTH 50
#define MAX_PENULIS_LENGTH 50

typedef struct Buku {
    int id;
    char judul[MAX_JUDUL_LENGTH];
    char penulis[MAX_PENULIS_LENGTH];
    int dipinjam; // 0: tersedia, 1: dipinjam
    struct Buku *next;
} Buku;

Buku *head = NULL;
int jumlahBuku = 0;

void tambahBuku() {
    Buku *bukuBaru = (Buku *)malloc(sizeof(Buku));
    if (!bukuBaru) {
        printf("Gagal mengalokasikan memori.\n");
        return;
    }

    bukuBaru->id = jumlahBuku + 1;
    printf("Masukkan judul: ");
    scanf(" %[^\\n]%*c", bukuBaru->judul); // Membaca string dengan spasi
    printf("Masukkan penulis: ");
    scanf(" %[^\\n]%*c", bukuBaru->penulis);
    bukuBaru->dipinjam = 0;
    bukuBaru->next = NULL;

    if (!head) {
        head = bukuBaru;
    } else {
        Buku *temp = head;
        while (temp->next) {
            temp = temp->next;
        }
        temp->next = bukuBaru;
    }

    jumlahBuku++;
    printf("Buku berhasil ditambahkan!\\n");
}
```

```

void pinjamBuku() {
    int id;
    printf("Masukkan ID buku yang ingin dipinjam: ");
    scanf("%d", &id);

    Buku *temp = head;
    while (temp) {
        if (temp->id == id) {
            if (temp->dipinjam) {
                printf("Buku sudah dipinjam.\n");
            } else {
                temp->dipinjam = 1;
                printf("Anda berhasil meminjam \"%s\".\n", temp->judul);
            }
            return;
        }
        temp = temp->next;
    }
    printf("Buku tidak ditemukan.\n");
}

void kembalikanBuku() {
    int id;
    printf("Masukkan ID buku yang ingin dikembalikan: ");
    scanf("%d", &id);

    Buku *temp = head;
    while (temp) {
        if (temp->id == id) {
            if (!temp->dipinjam) {
                printf("Buku tidak dipinjam.\n");
            } else {
                temp->dipinjam = 0;
                printf("Anda berhasil mengembalikan \"%s\".\n", temp->judul);
            }
            return;
        }
        temp = temp->next;
    }
    printf("Buku tidak ditemukan.\n");
}

void cariBuku() {
    char kataKunci[MAX_JUDUL_LENGTH];
    printf("Masukkan judul atau penulis untuk mencari: ");
    scanf(" %[^\n]*c", kataKunci);

    Buku *temp = head;

```

```

while (temp) {
    if (strstr(temp->judul, kataKunci) != NULL || strstr(temp->penulis, kataKunci) !=
NULL) {
        printf("ID: %d, Judul: %s, Penulis: %s, %s\n", temp->id, temp->judul, temp-
>penulis,
            temp->dipinjam ? "Dipinjam" : "Tersedia");
    }
    temp = temp->next;
}
}

```

```

void daftarBuku() {
    printf("\nDaftar buku di perpustakaan:\n");
    Buku *temp = head;
    while (temp) {
        printf("ID: %d, Judul: %s, Penulis: %s, %s\n", temp->id, temp->judul, temp-
>penulis,
            temp->dipinjam ? "Dipinjam" : "Tersedia");
        temp = temp->next;
    }
}

```

```

void tukarBuku(Buku *a, Buku *b) {
    int idTemp = a->id;
    char judulTemp[MAX_JUDUL_LENGTH];
    char penulisTemp[MAX_PENULIS_LENGTH];
    int dipinjamTemp = a->dipinjam;

    strcpy(judulTemp, a->judul);
    strcpy(penulisTemp, a->penulis);

    a->id = b->id;
    strcpy(a->judul, b->judul);
    strcpy(a->penulis, b->penulis);
    a->dipinjam = b->dipinjam;

    b->id = idTemp;
    strcpy(b->judul, judulTemp);
    strcpy(b->penulis, penulisTemp);
    b->dipinjam = dipinjamTemp;
}

```

```

void sortBuku() {
    if (!head || !head->next) {
        return;
    }

    int swapped;
    Buku *ptr1;

```

```

Buku *lptr = NULL;

do {
    swapped = 0;
    ptr1 = head;

    while (ptr1->next != lptr) {
        if (strcmp(ptr1->judul, ptr1->next->judul) > 0) {
            tukarBuku(ptr1, ptr1->next);
            swapped = 1;
        }
        ptr1 = ptr1->next;
    }
    lptr = ptr1;
} while (swapped);

printf("Buku berhasil diurutkan berdasarkan judul.\n");
}

int main() {
    int pilihan;

    while (1) {
        printf("\nSistem Manajemen Perpustakaan\n");
        printf("1. Tambah Buku\n");
        printf("2. Pinjam Buku\n");
        printf("3. Kembalikan Buku\n");
        printf("4. Cari Buku\n");
        printf("5. Daftar Semua Buku\n");
        printf("6. Urutkan Buku Berdasarkan Judul\n");
        printf("7. Keluar\n");
        printf("Masukkan pilihan Anda: ");
        scanf("%d", &pilihan);

        switch (pilihan) {
            case 1:
                tambahBuku();
                break;
            case 2:
                pinjamBuku();
                break;
            case 3:
                kembalikanBuku();
                break;
            case 4:
                cariBuku();
                break;
            case 5:
                daftarBuku();

```

```

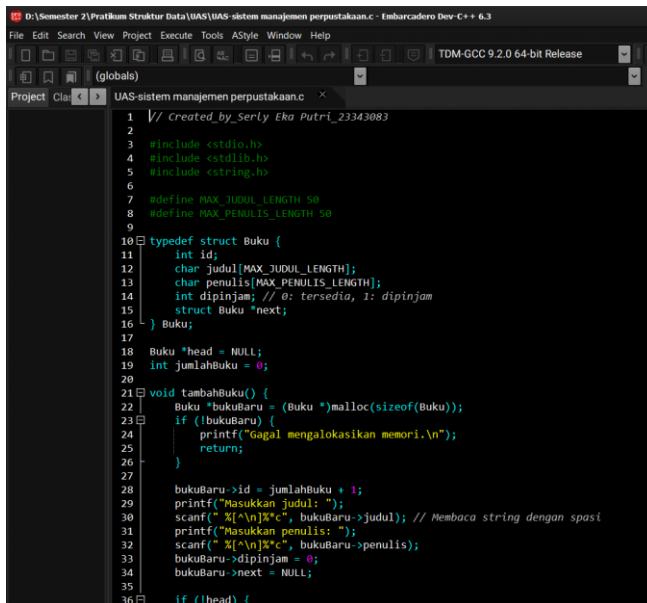
        break;
    case 6:
        sortBuku();
        break;
    case 7:
        exit(0);
    default:
        printf("Pilihan tidak valid!\n");
    }
}

return 0;
}

```

### C. Hasilnya / Screenshot

#### Screenshot Code



```

1  // Created by Serly Eka Putri_23343083
2
3  #include <stdio.h>
4  #include <stdlib.h>
5  #include <string.h>
6
7  #define MAX_JUDUL_LENGTH 50
8  #define MAX_PENULIS_LENGTH 50
9
10 typedef struct Buku {
11     int id;
12     char judul[MAX_JUDUL_LENGTH];
13     char penulis[MAX_PENULIS_LENGTH];
14     int dipinjam; // 0: tersedia, 1: dipinjam
15     struct Buku *next;
16 } Buku;
17
18 Buku *head = NULL;
19 int jumlahBuku = 0;
20
21 void tambahBuku() {
22     Buku *bukuBaru = (Buku *)malloc(sizeof(Buku));
23     if (!bukuBaru) {
24         printf("Gagal mengalokasikan memori.\n");
25         return;
26     }
27
28     bukuBaru->id = jumlahBuku + 1;
29     printf("Masukkan judul: ");
30     scanf("%[^\n]s", bukuBaru->judul); // Membaca string dengan spasi
31     printf("Masukkan penulis: ");
32     scanf("%[^\n]s", bukuBaru->penulis);
33     bukuBaru->dipinjam = 0;
34     bukuBaru->next = NULL;
35
36     if (!head) {

```

```
D:\Semester 2\Pratikum Struktur Data\UAS\UAS-sistem manajemen perpustakaan.c - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project Cla UAS-sistem manajemen perpustakaan.c
34 bukuBaru->next = NULL;
35
36 if (!head) {
37     head = bukuBaru;
38 } else {
39     Buku *temp = head;
40     while (temp->next) {
41         temp = temp->next;
42     }
43     temp->next = bukuBaru;
44 }
45
46 jumlahBuku++;
47 printf("Buku berhasil ditambahkan!\n");
48 }
49
50 void pinjamBuku() {
51     int id;
52     printf("Masukkan ID buku yang ingin dipinjam: ");
53     scanf("%d", &id);
54
55     Buku *temp = head;
56     while (temp) {
57         if (temp->id == id) {
58             if (temp->dipinjam) {
59                 printf("Buku sudah dipinjam.\n");
60             } else {
61                 temp->dipinjam = 1;
62                 printf("Anda berhasil meminjam \"%s\".\n", temp->judul);
63             }
64             return;
65         }
66         temp = temp->next;
67     }
68     printf("Buku tidak ditemukan.\n");
69 }
```

```
D:\Semester 2\Pratikum Struktur Data\UAS\UAS-sistem manajemen perpustakaan.c - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project Cla UAS-sistem manajemen perpustakaan.c
70
71 void kembalikanBuku() {
72     int id;
73     printf("Masukkan ID buku yang ingin dikembalikan: ");
74     scanf("%d", &id);
75
76     Buku *temp = head;
77     while (temp) {
78         if (temp->id == id) {
79             if (!temp->dipinjam) {
80                 printf("Buku tidak dipinjam.\n");
81             } else {
82                 temp->dipinjam = 0;
83                 printf("Anda berhasil mengembalikan \"%s\".\n", temp->judul);
84             }
85             return;
86         }
87         temp = temp->next;
88     }
89     printf("Buku tidak ditemukan.\n");
90 }
91
92 void cariBuku() {
93     char kataKunci[MAX_JUDUL_LENGTH];
94     printf("Masukkan judul atau penulis untuk mencari: ");
95     scanf("%[^\n]*c", kataKunci);
96
97     Buku *temp = head;
98     while (temp) {
99         if (strstr(temp->judul, kataKunci) != NULL || strstr(temp->penulis, kataKunci) != NULL) {
100             printf("ID: %d, Judul: %s, Penulis: %s, %s\n", temp->id, temp->judul, temp->penulis,
101                 temp->dipinjam ? "Dipinjam" : "Tersedia");
102         }
103         temp = temp->next;
104     }
105 }
```



```

D:\Semester 2\Pratikum Struktur Data\UAS\UAS-sistem manajemen perpustakaan.c - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project Cla UAS-sistem manajemen perpustakaan.c
106
107 void daftarBuku() {
108     printf("\ndaftar buku di perpustakaan:\n");
109     Buku *temp = head;
110     while (temp) {
111         printf("ID: %d, Judul: %s, Penulis: %s, %s\n", temp->id, temp->judul, temp->penulis,
112             temp->dipinjam ? "Dipinjam" : "Tersedia");
113         temp = temp->next;
114     }
115 }
116
117 void tukarBuku(Buku *a, Buku *b) {
118     int idTemp = a->id;
119     char judulTemp[MAX_JUDUL_LENGTH];
120     char penulisTemp[MAX_PENULIS_LENGTH];
121     int dipinjamTemp = a->dipinjam;
122
123     strcpy(judulTemp, a->judul);
124     strcpy(penulisTemp, a->penulis);
125
126     a->id = b->id;
127     strcpy(a->judul, b->judul);
128     strcpy(a->penulis, b->penulis);
129     a->dipinjam = b->dipinjam;
130
131     b->id = idTemp;
132     strcpy(b->judul, judulTemp);
133     strcpy(b->penulis, penulisTemp);
134     b->dipinjam = dipinjamTemp;
135 }
136
137 void sortBuku() {
138     if (!head || !head->next) {
139         return;
140     }
141

```

```

D:\Semester 2\Pratikum Struktur Data\UAS\UAS-sistem manajemen perpustakaan.c - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project Cla UAS-sistem manajemen perpustakaan.c
136
137 void sortBuku() {
138     if (!head || !head->next) {
139         return;
140     }
141
142     int swapped;
143     Buku *ptr1;
144     Buku *lptr = NULL;
145
146     do {
147         swapped = 0;
148         ptr1 = head;
149
150         while (ptr1->next != lptr) {
151             if (strcmp(ptr1->judul, ptr1->next->judul) > 0) {
152                 tukarBuku(ptr1, ptr1->next);
153                 swapped = 1;
154             }
155             ptr1 = ptr1->next;
156         }
157         lptr = ptr1;
158     } while (swapped);
159
160     printf("Buku berhasil diurutkan berdasarkan judul.\n");
161 }
162
163 int main() {
164     int pilihan;
165
166     while (1) {
167         printf("\nSistem Manajemen Perpustakaan\n");
168         printf("1. Tambah Buku\n");
169         printf("2. Pinjam Buku\n");
170         printf("3. Kembalikan Buku\n");
171         printf("4. Cari Buku\n");

```

```
D:\Semester 2\Pratikum Struktur Data\UAS\UAS-sistem manajemen perpustakaan.c - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project: UAS-sistem manajemen perpustakaan.c
166 while (1) {
167     printf("\nSistem Manajemen Perpustakaan\n");
168     printf("1. Tambah Buku\n");
169     printf("2. Pinjam Buku\n");
170     printf("3. Kembalikan Buku\n");
171     printf("4. Cari Buku\n");
172     printf("5. Daftar Semua Buku\n");
173     printf("6. Urutkan Buku Berdasarkan Judul\n");
174     printf("7. Keluar\n");
175     printf("Masukkan pilihan Anda: ");
176     scanf("%d", &pilihan);
177
178     switch (pilihan) {
179         case 1:
180             tambahBuku();
181             break;
182         case 2:
183             pinjamBuku();
184             break;
185         case 3:
186             kembalikanBuku();
187             break;
188         case 4:
189             cariBuku();
190             break;
191         case 5:
192             daftarBuku();
193             break;
194         case 6:
195             sortBuku();
196             break;
197         case 7:
198             exit(0);
199         default:
200             printf("Pilihan tidak valid!\n");
201     }
```

```
D:\Semester 2\Pratikum Struktur Data\UAS\UAS-sistem manajemen perpustakaan.c - Embarcadero Dev-C++ 6.3
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 9.2.0 64-bit Release
(globals)
Project: UAS-sistem manajemen perpustakaan.c
172     printf("5. Daftar Semua Buku\n");
173     printf("6. Urutkan Buku Berdasarkan Judul\n");
174     printf("7. Keluar\n");
175     printf("Masukkan pilihan Anda: ");
176     scanf("%d", &pilihan);
177
178     switch (pilihan) {
179         case 1:
180             tambahBuku();
181             break;
182         case 2:
183             pinjamBuku();
184             break;
185         case 3:
186             kembalikanBuku();
187             break;
188         case 4:
189             cariBuku();
190             break;
191         case 5:
192             daftarBuku();
193             break;
194         case 6:
195             sortBuku();
196             break;
197         case 7:
198             exit(0);
199         default:
200             printf("Pilihan tidak valid!\n");
201     }
202 }
203
204 return 0;
205 }
206
```

## Screenshoot hasil

```
DiSemester 2\Pratikum Struk x + v

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 1
Masukkan judul: filosofi
Masukkan penulis: buya hamka
Buku berhasil ditambahkan!

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 1
Masukkan judul: fisika
Masukkan penulis: ibnu sina
Buku berhasil ditambahkan!

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 1
Masukkan judul: organisasi arsitektur dan komputer
Masukkan penulis: ahmad dahlan
Buku berhasil ditambahkan!

Sistem Manajemen Perpustakaan
```

```
DiSemester 2\Pratikum Struk x + v

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 2
Masukkan ID buku yang ingin dipinjam: 3
Anda berhasil meminjam "organisasi arsitektur dan komputer".

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 2
Masukkan ID buku yang ingin dipinjam: 1
Anda berhasil meminjam "filosofi".

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 2
Masukkan ID buku yang ingin dipinjam: 2
Anda berhasil meminjam "fisika ".

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
```

```
D:\Semester 2\Pratikum Struk x + v
Anda berhasil meminjam "fisika ".

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 3
Masukkan ID buku yang ingin dikembalikan: 1
Anda berhasil mengembalikan "filosofi".

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 3
Masukkan ID buku yang ingin dikembalikan: 2
Anda berhasil mengembalikan "fisika ".

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 3
Masukkan ID buku yang ingin dikembalikan: 3
Anda berhasil mengembalikan "organisasi arsitektur dan komputer".

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
```

```
D:\Semester 2\Pratikum Struk x + v
Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 4
Masukkan judul atau penulis untuk mencari: ahmad dahlan
ID: 3, Judul: organisasi arsitektur dan komputer, Penulis: ahmad dahlan, Tersedia

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 4
Masukkan judul atau penulis untuk mencari: filosofi
ID: 1, Judul: filosofi, Penulis: buya hanka, Tersedia

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 4
Masukkan judul atau penulis untuk mencari: fisika
ID: 2, Judul: fisika , Penulis: ibnu sina, Tersedia

Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
```

```
D:\Semester 2\Pratikum Struk x + v
Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 5
Daftar buku di perpustakaan:
ID: 1, Judul: filosofi, Penulis: buya hanka, Tersedia
ID: 2, Judul: fisika, Penulis: ibnu sina, Tersedia
ID: 3, Judul: organisasi arsitektur dan komputer, Penulis: ahmad dahlan, Tersedia
Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 6
Buku berhasil diurutkan berdasarkan judul.
Sistem Manajemen Perpustakaan
1. Tambah Buku
2. Pinjam Buku
3. Kembalikan Buku
4. Cari Buku
5. Daftar Semua Buku
6. Urutkan Buku Berdasarkan Judul
7. Keluar
Masukkan pilihan Anda: 7
-----
Process exited after 260.8 seconds with return value 0
Press any key to continue . . . |
```

#### D. Penjelasan Program

```
2
3 #include <stdio.h>
4 #include <stdlib.h>
5 #include <string.h>
6
```

1. `#include <stdio.h>`: Mengimpor Pustaka standar input-output untuk fungsi seperti 'printf' dan 'scanf'.
2. `#include <stdlib.h>`: Mengimpor Pustaka standar untuk fungsi seperti 'malloc' dan 'free'.
3. `#include <string.h>`: Mengimpor Pustaka string untuk fungsi seperti 'strcpy' dan 'strchr'

```
6
7 // Definisi panjang maksimal judul dan penulis buku
8 #define MAX_JUDUL_LENGTH 50
9 #define MAX_PENULIS_LENGTH 50
10
```

4. `#define MAX_JUDUL_LENGTH 50`: Mendefinisikan Panjang maksimum untuk judul buku sebagai 50 karakter.
5. `#define MAX_PENULIS_LENGTH 50`: Mendefinisikan Panjang maksimum untuk penulis buku sebagai 50 karakter.

```

10
11 // Definisi struktur Buku
12 typedef struct Buku {
13     int id; // ID buku
14     char judul[MAX_JUDUL_LENGTH]; // Judul buku
15     char penulis[MAX_PENULIS_LENGTH]; // Penulis buku
16     int dipinjam; // Status peminjaman (0: tersedia, 1: dipinjam)
17     struct Buku *next; // Pointer ke buku berikutnya dalam linked list
18 } Buku;
19

```

6. `typedef struct buku {...} buku;` : Mendefinisikan struktur 'Buku' dengan beberapa atribut: 'id', 'judul', 'penulis', 'dipinjam', dan pointer 'next' yang menunjuk ke buku berikutnya dalam linked list.
7. `'Buku *head = NULL;` : Mendefinisikan pointer 'head' yang menunjukkan ke buku pertama dalam linked list.
8. `'Buku *head = NULL;` : Mendefinisikan pointer 'head' yang menunjukkan ke buku pertama dalam linked list.
9. `' int jumlahBuku = 0;` : Mendefinisikan variabel jumlahBuku untuk menyimpan jumlah total buku dalam perpustakaan.
10. `void tambahBuku() { /* ... */ }` : Mendefinisikan fungsi tambahBuku() untuk menambahkan buku baru ke dalam perpustakaan.
11. `void pinjamBuku() { /* ... */ }` : Mendefinisikan fungsi pinjamBuku() untuk meminjam buku dari perpustakaan.
12. `void kembalikanBuku() { /* ... */ }` : Mendefinisikan fungsi kembalikanBuku() untuk mengembalikan buku yang sudah dipinjam.
13. `void cariBuku() { /* ... */ }` : Mendefinisikan fungsi cariBuku() untuk mencari buku berdasarkan judul atau penulis.
14. `void daftarBuku() { /* ... */ }` : Mendefinisikan fungsi daftarBuku() untuk menampilkan daftar semua buku dalam perpustakaan.
15. `void tukarBuku(Buku a, Buku *b) { / ... */ }` : Mendefinisikan fungsi tukarBuku() untuk menukar informasi antara dua buku.
16. `void sortBuku() { /* ... */ }` : Mendefinisikan fungsi sortBuku() untuk mengurutkan buku dalam perpustakaan berdasarkan judul.
17. `int main() { /* ... */ }` : Mendefinisikan fungsi utama main() yang akan menjalankan program.
18. `int pilihan;` : Mendeklarasikan variabel pilihan untuk menyimpan pilihan menu dari pengguna.
19. `while (1) { /* ... */ }` : Memulai loop tak terbatas untuk menampilkan menu dan memproses pilihan pengguna.
20. `printf("\nSistem Manajemen Perpustakaan\n") ;` : Menampilkan judul program.
21. `scanf("%d", &pilihan) ;` : Meminta pengguna untuk memasukkan pilihan menu.
22. `switch (pilihan) { /* ... */ }` : Memulai struktur switch-case untuk memproses pilihan pengguna.
23. `case 1: tambahBuku(); break;` : Memanggil fungsi tambahBuku() jika pengguna memilih opsi untuk menambahkan buku.
24. `case 2: pinjamBuku(); break;` : Memanggil fungsi pinjamBuku() jika pengguna memilih opsi untuk meminjam buku.
25. `case 3: kembalikanBuku(); break;` : Memanggil fungsi kembalikanBuku() jika pengguna memilih opsi untuk mengembalikan buku.

26. `case 4: cariBuku(); break;` Memanggil fungsi `cariBuku()` jika pengguna memilih opsi untuk mencari buku.
27. `case 5: daftarBuku(); break;` Memanggil fungsi `daftarBuku()` jika pengguna memilih opsi untuk menampilkan daftar semua buku.
28. `case 6: sortBuku(); break;` Memanggil fungsi `sortBuku()` jika pengguna memilih opsi untuk mengurutkan buku berdasarkan judul.

## E. Kesimpulan

System manajemen perpustakaan yang diimplementasikan dalam program ini menyediakan antarmuka sederhana untuk pengelolaan buku dalam perpustakaan.

Berikut ini adalah beberapa Kesimpulan dari system ini :

1. Kemudahan pengguna :  
Sistem menyediakan menu sederhana yang memungkinkan pengguna dengan mudah melakukan operasi seperti menambahkan buku baru, meminjam buku, mengembalikan buku, mencari buku berdasarkan judul atau penulis, menampilkan daftar semua buku, dan mengurutkan buku berdasarkan judul.
2. Manajemen buku :  
Setiap buku direpresentasikan sebagai sebuah entitas dengan atribut seperti judul, penulis, ID, dan status peminjaman. Hal ini memungkinkan sistem untuk melakukan operasi-operasi yang diperlukan seperti meminjam dan mengembalikan buku dengan mudah.
3. Pencarian Efisien :  
Sistem memungkinkan pengguna untuk mencari buku berdasarkan judul atau penulis dengan memasukkan kata kunci pencarian. Ini memberikan kemudahan bagi pengguna untuk menemukan buku yang mereka cari.
4. Tampilan Informasi :  
Fungsi `daftarBuku()` memberikan daftar semua buku dalam perpustakaan beserta informasi penting seperti judul, penulis, ID, dan status peminjaman. Ini memungkinkan pengguna untuk melihat informasi buku dengan mudah.
5. Pengurutan Buku :  
Sistem dapat mengurutkan buku dalam perpustakaan berdasarkan judul secara alfabetis menggunakan fungsi `sortBuku()`. Ini membantu pengguna dalam menavigasi dan menemukan buku dengan lebih efisien.
6. Kontrol Kesalahan :  
Sistem memiliki mekanisme untuk menangani kesalahan seperti gagal alokasi memori saat menambahkan buku baru atau meminjam buku yang tidak ada dalam perpustakaan.

Dengan demikian, Kesimpulan dari system manajemen perpustakaan ini adalah menyediakan antarmuka yang mudah digunakan, efisien dalam pencarian dan manajemen buku, serta memberikan fungsionalitas dasar yang diperlukan dalam sebuah perpustakaan. System manajemen perpustakaan ini menyediakan Solusi yang efisien, dapat diandalkan, dan mudah digunakan untuk mengelola perpustakaan dengan baik. Sytem ini dapat membantu meningkatkan efisiensi dan keteraturan dalam pengelolaan perpustakaan.