KELOMPOK 5 STRUKTUR DATA

PENDATAAN KOMIK PADA GUDANG

LINKED LIST & QUEUE C++

ANGGOTA KELOMPOK 5



AZKA AVICENNA RASJID 20081010115



FARKHAN

20081010060



KUNCORO ARIADI

20081010096

TENTANG PROGRAM

PROGRAM INI DIBUAT UNTUK
MELAKUKAN PENDATAAN KOMIK YANG
DISIMPAN DI DALAM GUDANG. KOMIKKOMIK INI DI SIMPAN DI GUDANG
SEBELUM DIKIRIMKAN KE TOKO DAN SIAP
UNTUK DIJUAL. KOMIK YANG DIKIRIMKAN
KE TOKO MERUPAKAN KOMIK YANG
MASUK KE GUDANG TERLEBIH DULU.

```
Program Pendataan Komik Pada Suatu Gudang ===
DISUSUN OLEH KELOMPOK 5
Azka Avicenna Rasjid
                         [20081010115]
Farkhan
                         20081010060
Kuncoro Ariadi
                         [20081010096]
Pilihan menu yang tersedia :
1. Tambah Komik
2. Kirim Komik ke Toko
3. Tampil Komik di Gudang
4. Cari Komik di Gudang
5. Hapus Komik di Gudang
6. Keluar
>> Masukkan Pilihan :
```

FUNGSI UTAMA

```
do {
            cout << "Pilihan menu yang tersedia : " << endl;</pre>
            cout << "1. Tambah Komik" << endl;</pre>
            cout << "2. Kirim Komik ke Toko" << endl;</pre>
            cout << "3. Tampil Komik di Gudang" << endl;</pre>
            cout << "4. Cari Komik di Gudang" << endl;</pre>
            cout << "5. Hapus Komik di Gudang" << endl;</pre>
            cout << "6. Keluar" << endl << endl;</pre>
            cout << ">> Masukkan Pilihan : ";
            cin >> pilihan;
 11
            cout << endl;</pre>
 12
            switch (pilihan) {
 13
                case 1: tambah(); getch(); break;
 14
                case 2: kirim(); getch(); break;
 15
                case 3: tampil(); getch(); break;
                case 4: cari(); getch(); break;
 17
                case 5: hapus(); getch(); break;
                case 6:
                     cout << "Terima Kasih Telah Menggunakan Program Ini..." << endl;</pre>
20
                     getch(); break;
 21
                default:
 22
                     cout << "Pilihan tidak tersedia" << endl;</pre>
23
24
            system("cls");
       } while (pilihan != 6);
        return 0;
 27 }
```

DEKLARASI STRUCT DAN VARIABLE

```
1 struct node {
      int kode_komik;
      char judul[50];
      char penerbit[50];
      char pengarang[50];
     int tahun;
      struct node *next;
8 };
9 struct node* front = NULL;
10 struct node* rear = NULL;
11 struct node* temp;
```

FUNGSI TAMBAH DATA

```
1 void tambah() {
2    int kode_komik, tahun;
3    char judul[50], penerbit[50], pengarang[50];
4
5    cout << "> Masukkan Kode Komik : "; cin >> kode_komik;
6    cout << "> Masukkan Judul Komik : "; cin >> judul;
7    cout << "> Masukkan Penerbit Komik : "; cin >> penerbit;
8    cout << "> Masukkan Pengarang Komik : "; cin >> pengarang;
9    cout << "> Masukkan Tahun Terbit Komik : "; cin >> tahun;
```

```
if (rear == NULL) {
           rear = (struct node *)malloc(sizeof(struct node));
           rear->next = NULL;
           rear->kode_komik = kode_komik;
           strcpy(rear->judul, judul);
           strcpy(rear->penerbit, penerbit);
           strcpy(rear->pengarang, pengarang);
           rear->tahun = tahun;
           front = rear;
10
       } else {
           temp=(struct node *)malloc(sizeof(struct node));
11
12
           rear->next = temp;
           temp->kode_komik = kode_komik;
13
           strcpy(temp->judul, judul);
14
           strcpy(temp->penerbit, penerbit);
15
16
           strcpy(temp->pengarang, pengarang);
           temp->tahun = tahun;
17
18
           temp->next = NULL;
19
           rear = temp;
20
       cout << "\nKomik Berhasil Ditambahkan" << endl << endl;</pre>
22 }
```

FUNGSI HAPUS URUTAN

```
1 void kirim() {
       temp = front;
       if (front == NULL) {
            cout << "Gudang kosong, masukkan data komik terlebih dahulu..." << endl;</pre>
            return;
       } else if (temp->next != NULL) {
            temp = temp->next;
            cout << "Komik yang dikirim ke toko adalah : " << front->judul << endl;</pre>
           free(front);
10
            front = temp;
11
       } else {
12
            cout << "Komik yang dikirim ke toko adalah : " << front->judul << endl;</pre>
13
            free(front);
           front = NULL;
14
15
            rear = NULL;
16
17 }
```

FUNGSI MENAMPILKAN DATA

```
1 void tampil() {
        temp = front;
        if ((front == NULL) && (rear == NULL)) {
            cout<<"Gudang kosong, masukkan data komik terlebih dahulu..."<<endl << endl;</pre>
            return;
 6
       int i = 1;
        cout << "Data komik yang tersedia : " << endl << endl;</pre>
 8
        while (temp != NULL) {
 9
            cout << "[ Data Komik Ke-" << i << " ]"<< endl;</pre>
10
            cout << "Kode Komik : " << temp->kode_komik << endl;</pre>
11
                                   : " << temp->judul << endl;</pre>
12
            cout << "Judul Komik
                                       : " << temp->penerbit << endl;</pre>
13
            cout << "Penerbit Komik</pre>
                                              : " << temp->pengarang << endl;</pre>
14
            cout << "Pengarang Komik</pre>
            cout << "Tahun Terbit Komik : " << temp->tahun << endl << endl;</pre>
15
16
            i++;
17
            temp = temp->next;
18
        cout<<endl;</pre>
19
20 }
```

FUNGSI CARI DATA

```
1 void cari() {
        int kode_komik, flag = 0;
       temp = front;
       if (temp == NULL) {
            cout << "Gudang kosong, masukkan data komik terlebih dahulu..." << endl;</pre>
            return;
10
       cout << "Masukkan Kode Komik : "; cin >> kode_komik;
11
       while (temp != NULL) {
12
            if (temp->kode_komik == kode_komik) { flag = 1; break; }
13
            temp = temp->next;
14
       }
15
16
       if (flag == 1) {
            cout << "\nKode Komik</pre>
17
                                         : " << temp->kode_komik << endl;</pre>
            cout << "Judul Komik : " << temp->judul << endl;</pre>
18
                                         : " << temp->penerbit << endl;</pre>
19
            cout << "Penerbit Komik</pre>
                                             : " << temp->pengarang << endl;</pre>
            cout << "Pengarang Komik</pre>
20
            cout << "Tahun Terbit Komik : " << temp->tahun << endl;</pre>
21
       } else { cout << "Kode Komik tidak ditemukan..." << endl; }</pre>
22
23
24
       cout << endl;</pre>
25 }
```

FUNGSI MENGHAPUS DATA

```
1 void hapus() {
       int kode_komik, int flag = 0;
       struct node *temp1;
       temp = front;
       if (temp == NULL) {
           cout << "Gudang kosong, masukkan data komik terlebih dahulu..." << endl;</pre>
10
       cout << "Masukkan Kode Komik : "; cin >> kode_komik;
11
       while (temp != NULL) {
12
           if (temp->kode_komik == kode_komik) {
13
               flag = 1;
14
               break;
15
           temp1 = temp;
17
           temp = temp->next;
```

```
if (flag == 1) {
           if (temp == front) {
               if (temp->next != NULL) {
                    temp = temp->next;
                   cout << "Komik yang dihapus adalah : " << front->judul << endl;</pre>
                    free(front);
                    front = temp;
               } else {
                    cout << "Komik yang dihapus adalah : " << front->judul << endl;</pre>
10
                    free(front);
11
                    front = NULL;
12
                    rear = NULL;
13
14
           } else if (temp->next == NULL) {
15
               rear = temp1;
16
                rear->next = NULL;
               cout << "Komik yang dihapus adalah : " << temp->judul << endl;</pre>
17
18
                free(temp);
19
           } else {
20
                temp1->next = temp->next;
               cout << "Komik yang dihapus adalah : " << temp->judul << endl;</pre>
21
22
                free(temp);
23
       } else {
24
25
           cout << "Kode Komik tidak ditemukan" << endl;</pre>
26
27
28
       cout << endl;</pre>
29 }
```

DEMO PROGRAM



TERIMA KASIH

