**LAPORAN ALGORITMA DAN PEMROGRAMAN PROGRAM BIOSKOP**



Oleh:

KELOMPOK 07

Putu Satria Arya Putra (2305551122)

Azka Ilham Ramadhani (2305551171)

Anthony Wisnu Jati (2305551138)

**PROGRAM STUDI TEKNOLOGI INFORMASI**

**FAKULTAS TEKNIK**

**UNIVERSITAS UDAYANA**

**BALI**

**2023**

# KATA PENGANTAR

Puji syukur kami panjatkan ke kehadirat Tuhan Yang Maha Esa, karena berkat rahmat-Nya serta dengan usaha, dan semangat penyusun, laporan ini dapat diselesaikan dengan tepat waktu. Penulisan laporan yang berjudul “Laporan Algoritma dan Pemrograman Program Bioskop”. Terima kasih dan penghargaan yang setinggi-tingginya disampaikan kepada pihak-pihak yang telah membantu dalam menyelesaikan laporan ini, yaitu sebagai berikut.

1. Bapak Kadek Suar Wibawa, S.T., M.T. selaku Dosen Pengampu Mata Kuliah Algoritma Pemrograman.

Kemudian, saya menyadari bahwa tugas yang saya tulis ini masih jauh dari kata sempurna. Oleh karena itu, kritik dan saran yang membangun kami butuhkan demi kesempurnaan laporan ini. Apabila terdapat sekiranya hal-hal yang kurang berkenan di benak pembaca dalam laporan ini, penyusun memohon maaf yang sebesar-besarnya. Akhir kata, semoga laporan Praktikum Algoritma Dan Pemrograman ini memberikan manfaat bagi pembaca. Atas perhatian para pembaca, penyusun mengucapkan terima kasih.

Jimbaran, Desember 2023

(Penyusun)

# ABSTRAK

Pokok bahasan pada laporani ini iadalah tentang ipemrograman menggunakan bahasa C/C++, dimana mencakup mulai dari dasar pemrograman hingga penerapan bahasa pemrograman dalam mengerjakan soal dengan studi kasus yang diberikan yakni membuat program bioskop. Algoritma merupakan metode (langkah) yang direncanakan secara tersusun dan berurut untuk menyelesaikan permasalahan dengan sebuah perintah, kegiatan atau rangkaian terbatas dari instruksi-instruksi yang rumit, yang biasanya digunakan untuk menyelesaikan atau menjalankan suatu kelompok masalah komputasi tertentu. Algoritma digunakan sebagai spesifikasi untuk melakukan perhitungan dan pemrosesan data. Selain digunakan untuk memecahkan masalah dengan menggunakan *computer*, algoritma juga dapat diterapkan dalam menyelesaikan permasalahan pada kehidupan sehari-hari yang memerlukan sederet proses atau langkah-langkah. Maka dapat kami simpulkan bahwa, konsep pemrograman ialah sebuah pernyataan yang tersusun membentuk suatu kesatuan prosedur berupa tahapan dan berfungsi iuntuk menyelesaikan ipermasalahan iyang adai dengan mengimplementasikan dengan membuat *flowchart* dan *pseudocode* serta pengaplikasiannya menggunakan bahasa pemrograman C/C++ sehingga bahasa tersebut bisa dimengerti dan kemudian dikerjakan oleh *compiler*. Kemudian hasil dari perancangan tersebut akan di *tracing* yang berfungsi untuk memastikan bahwa kode program yang dibuat sudah berjalan benar.

# DAFTAR ISI

[KATA PENGANTAR ii](#_Toc154083658)

[ABSTRAK iii](#_Toc154083659)

[DAFTAR ISI iv](#_Toc154083660)

[BAB I PENDAHULUAN 1](#_Toc154083661)

[1.1 Latar Belakang 1](#_Toc154083662)

[1.2 Rumusan Masalah 1](#_Toc154083663)

[1.3 Tujuan 1](#_Toc154083664)

[BAB II PEMBAHASAN 2](#_Toc154083665)

[2.1 Flowchart Program Bioskop 2](#_Toc154083666)

[2.2 Pseudocode Program Bioskop 10](#_Toc154083667)

[2.3 Kode Program Bioskop 16](#_Toc154083668)

[BAB III PENUTUP 32](#_Toc154083669)

[3.1 Simpulan 32](#_Toc154083670)

[3.2 Saran 32](#_Toc154083671)

# 

# BAB I PENDAHULUAN

## Latar Belakang

Munculnya teknologi informasi yang canggih serta internet, telah mendorong manusia menjadi lebih efektif dan efisien. Teknologi informasi merupakan teknologi yang berfungsi dalam mengolah dan memproses data dengan berbagai cara untuk mendapatkan informasi. Teknologi ini idigunakan untuk membuat, mengubah, imenyimpan, dan menyebarkan berbagai informasi. Didalam Teknologi informasi terdapat sebuah istilah *programming*.

Berdasarkan hal tersebut, dalam mata kuliah Algoritma dan Pemrograman bertujuan untuk melatih para mahasiswa supaya mampu menerapkan algoritma sederhana ke dalam pemrograman sehingga kita bisa mengatasi suatu masalah dalam sebuah persoalan yang ditentukan.

## Rumusan Masalah

Berdasarkan penjelasan latar belakang yang sudah dijelaskan di atas, maka dalam mata kuliah Algoritma Pemrograman ini dapat dirumuskan sebagai berikut.

1. Membuat sebuah program bioskop yang mencakup: login sebagai admin, login sebagai pelanggan, pemesanan tiket, kursi, dan film yang ingin ditonton.

## Tujuan

Berdasarkan penjelasan persoalan di atas, maka tujuan yang diharapkan dapat dicapai melalui mata kuliah Algoritma Pemrograman ini sebagai berikut.

1. Mampu membuat program bioskop yang mencakup: login sebagai admin, login sebagai pelanggan, pemesanan tiket, kursi, dan film yang ingin ditonton.

# BAB II PEMBAHASAN

Pada Bab II ini berisikan tentang hasil dan pembahasan terkait dengan *flowchart*, *pseudocode*, dan kode program. Adapun hasil dan pembahasan dibahas sebagai berikut.

## Flowchart Program Bioskop

*Flowchart* merupakan penggambaran visual dari alur atau proses program yang akan dibuat. Dalam pembuatan program bioskop ini *flowchart* digunakan untuk menentukan dan mendiskusikan algoritma awal dari program bioskop yang akan dibuat. Berikut merupakan *flowchart* dari program bioskop.

|  |
| --- |
|  |

**Gambar 2. 1** Flowchart Fungsi Pertama

Gambar 2.1 merupakan *flowchart* fungsi pertama dari program bioskop. Pada *flowchart* tersebut pengguna diminta untuk memasukkan angka 1 sampai 3. Dimana jika pengguna memasukkan angka 1, maka akan menuju fungsi login\_admin, jika memasukkan angka 2, maka menuju fungsi menu\_customer, jika angka 3 maka akan menyelesaikan program.

|  |
| --- |
|  |

**Gambar 2. 2** Flowchart Fungsi login\_admin

|  |
| --- |
|  |

**Gambar 2. 3** Flowchart Fungsi menu\_admin

Gambar 2.2 merupakan *flowchart* fungsi login\_admin, gambar 2.3 merupakan *flowchart* fungsi menu\_admin. Gambar 2.2 didapat jika pengguna memasukkan angka 1 pada fungsi pertama. Gambar 2.3 didapat jika pengguna sudah berhasil melakukan login pada fungsi login\_admin.

|  |
| --- |
|  |

**Gambar 2. 4** Flowchart Fungsi menu\_customer

Gambar 2.4 merupakan *flowchart* fungsi menu\_customer pada program bioskop. Fungsi tersebut didapat apabila pengguna memasukkan angka 2 pada fungsi pertama. Pada *flowchart* tersebut pengguna diminta untuk memasukkan angka 1 sampai 3. Dimana jika pengguna memasukkan angka 1, maka akan menuju fungsi registrasi, jika memasukkan angka 2, maka menuju fungsi login, jika angka 3 maka akan menyelesaikan program.

|  |
| --- |
|  |

**Gambar 2. 5** FlowchartFungsi Pemesanan

Gambar 2.5 merupakan *flowchart* fungsi pemesanan pada program bioskop. Fungsi ini didapat setelah pengguna melakukan registrasi atau login pada menu\_pelanggan. *Flowchart* diatas meminta pengguna memasukkan id film, jumlah tiket yang ingin dipesan, jam tayang, dan no kursi untuk menonton film tersebut.

|  |
| --- |
|  |

**Gambar 2. 6** Flowchart Fungsi Pembayaran

Gambar 2.6 merupakan *flowchart* fungsi pembayaran yang terdapat pada program bioskop. Fungsi pembayaran didapat apabila pengguna sudah melakukan pemesanan. Fungsi pembayaran berisi total harga dan pilihan untuk mencetak nota pembayaran.

## Pseudocode Program Bioskop

*Pseudocode* merupakan suatu deskripsi algoritma yang menyerupai kode program. *Pseudocode* dapat digunakan untuk memberikan gambaran algoritma yang nantinya akan diimplementasikan kedalam kode program secara langsung. Berikut merupakan *pseudocode* program bioskop.

|  |
| --- |
| procedure main():  call displayGroupMembers()  call showMenu()  procedure displayGroupMembers():  output "=================================================="  output "|| Nama Kelompok ||"  output "=================================================="  output "|| 1. Putu Satria Arya Putra (2305551122) ||"  output "|| 2. Anthony Wisnu Jati (2305551138) ||"  output "|| 3. Azka Ilham Ramadhani (2305551171) ||"  output "=================================================="  output "Enter untuk melanjutkan..."  procedure showMenu():  input choice  switch choice:  case 1:  call loginAdmin()  case 2:  call showCustomerMenu()  case 3:  call exitProgram()  default:  output "Error"  procedure loginAdmin():  input username  input password  if (username == "admin" and password == "123"):  call menuAdmin()  else:  output "Password atau Username Salah!"  output "Tekan Enter untuk Kembali..."  wait for Enter key press  call showMenu()  procedure menuAdmin():  repeat:  output "=============================================================="  output "|| Tampilan Menu Admin ||"  output "||==========================================================||"  output "|| [1] Menampilkan riwayat transaksi ||"  output "||----------------------------------------------------------||"  output "|| [2] Log out ||"  output "=============================================================="  output "Masukkan pilihan Anda [1][2][3] : "  input choice  switch choice:  case 1:  call displayTransactionHistory()  case 2:  call showMenu()  default:  output "Menu tidak tersedia!"  procedure displayTransactionHistory():  open "RiwayatTransaksi.txt" file  read and output content of the file  close file  output "Apakah Mau kembali ke menu admin ? (Y/N) : "  input decision  if (decision == "Y" or decision == "y"):  call menuAdmin()  else:  call exitProgram()  procedure showCustomerMenu():  output "=============================================================="  output "|| SELAMAT DATANG ||"  output "||==========================================================||"  output "|| [1] Registrasi ||"  output "||----------------------------------------------------------||"  output "|| [2] Login ||"  output "||----------------------------------------------------------||"  output "|| [3] Exit ||"  output "=============================================================="  output "Masukkan Pilihan Anda : "  input choice  if (choice == 1):  call registerCustomer()  elif (choice == 2):  call loginCustomer()  elif (choice == 3):  call exitProgram()  else:  output "Angka yang Anda inputkan salah!"  output "Mohon inputkan pilihan yang benar..."  output "Masukkan pilihan Anda: "  repeat:  input choice  until (choice is a valid option)  call showCustomerMenu()  procedure registerCustomer():  open "DataCustomer.txt" file for append  input username  while (username already exists in the file):  output "username sudah terdaftar. Coba gunakan username lainnya"  input new username  input PIN  write username and PIN to the file  close file  output "Registrasi berhasil!"  call loginCustomer()  procedure loginCustomer():  open "DataCustomer.txt" file for read  input username  input PIN  while (not end of file):  read username and PIN from the file  if (username and PIN match):  output "Login berhasil"  call menuOrder()  close file  output "Login gagal. Silahkan periksa username dan PIN Anda."  call showCustomerMenu()  procedure menuOrder():  output "==============================================="  output "|| SELAMAT DATANG DIBIOSKOP TI ||"  output "==============================================="  output "|| Tampilan Menu ||"  output "==============================================="  output "|| 1. Menu Pemesanan Tiket ||"  output "|| 2. Exit ||"  output "==============================================="  output "Masukkan Pilihan Anda: "  input choice  if (choice == 1):  call chooseMovieGenre()  elif (choice == 2):  call exitProgram()  else:  output "Error"  procedure chooseMovieGenre():  output "========================="  output "|| Genre Film ||"  output "========================="  output "|| 1. Horor ||"  output "|| 2. Komedi ||"  output "|| 3. Romantis ||"  output "========================="  output "Masukkan Pilihan Genre Film Anda: "  input genre  if (genre == 1):  call displayHorrorMovies()  elif (genre == 2):  call displayComedyMovies()  elif (genre == 3):  call displayRomanticMovies()  else:  output "Masukkan Pilihan Yang Benar"  procedure displayHorrorMovies():  define array of horror movies  call displayMovieList(horror movies)  procedure displayComedyMovies():  define array of comedy movies  call displayMovieList(comedy movies)  procedure displayRomanticMovies():  define array of romantic movies  call displayMovieList(romantic movies)  procedure displayMovieList(movies):  output "====================================================================================================="  output "|| Daftar Film yang Tayang ||"  output "====================================================================================================="  output " ID \t| Title \t\t| Genre \t| Stock \t| Price Weekday | Price Weekend"  output "-----------------------------------------------------------------------------------------------------"  for each movie in movies:  output movie details  output "Silahkan pilih ID film yang akan dipesan: "  input movieID  if (movieID is valid):  call performTicketReservation(selected movie)  else:  output "Maaf, ID film yang Anda masukkan tidak valid!"  procedure performTicketReservation(selectedMovie):  determine current day  determine movie price based on current day  output "Film yang Anda pilih adalah '{selectedMovie.title}'. Harga tiketnya adalah Rp {selectedMovie.price}"  output "Berapa jumlah tiket yang ingin Anda pesan: "  input ticketQuantity  if (ticketQuantity > selectedMovie.stock):  output "Maaf, stock tiket yang tersedia adalah {selectedMovie.stock}"  else:  reduce selectedMovie.stock by ticketQuantity  call selectShowTime(selectedMovie, ticketQuantity)  procedure selectShowTime(selectedMovie, ticketQuantity):  output "Daftar jam tayang:"  if (selectedMovie.price is weekday price):  output show weekday show times  else:  output show weekend show times  output "Silahkan Masukkan Jam tayang : "  input selectedShowTime  call selectSeats(selectedMovie, ticketQuantity, selectedShowTime)  procedure selectSeats(selectedMovie, ticketQuantity, selectedShowTime):  define array of available seats  call displaySeatMap()  output "Silahkan pilih nomor kursi (pisahkan dengan koma jika lebih dari satu): "  input selectedSeats  if (selectedSeats are available):  call confirmReservation(selectedMovie, ticketQuantity, selectedShowTime, selectedSeats)  else:  output "Maaf, kursi yang Anda pilih tidak tersedia. Silakan pilih kursi lain."  call selectSeats(selectedMovie, ticketQuantity, selectedShowTime)  procedure displaySeatMap():  output "Diagram Kursi Bioskop"  procedure confirmReservation(selectedMovie, ticketQuantity, selectedShowTime, selectedSeats):  output "Ringkasan Pemesanan:"  output "Film: {selectedMovie.title}"  output "Tanggal dan Jam: {selectedShowTime}"  output "Jumlah Tiket: {ticketQuantity}"  output "Nomor Kursi: {selectedSeats}"  output "Harga Total: Rp {ticketQuantity \* selectedMovie.price}"  output "Apakah Anda yakin ingin melanjutkan pemesanan? (Y/N): "  input confirmation  if (confirmation == "Y" or confirmation == "y"):  write reservation details to "TransactionHistory.txt"  output "Pemesanan berhasil! Terima kasih."  call menuOrder()  else:  call menuOrder()  procedure exitProgram():  output "Terima kasih telah menggunakan layanan kami. Sampai jumpa!"  exit program |

**Kode Program 2. 1** Pseudocode Program Bioskop

Kode Program 2.1 merupakan *pseudocode* dari program bioskop. Pada *pseudocode* program bioskop ini sudah mengimplementasikan alur proses dari algoritma yang sudah digambarkan dengan *flowchart* sebelumnya. Penulisan dari *psudocode* ini pun sudah menyerupai dari *syntax* dari bahasa pemrograman yang akan dibuat.

## Kode Program Bioskop

Program bioskop merupakan program untuk memesan tiket bioskop yang mencakup: login sebagai admin, login sebagai pelanggan, pemesanan tiket, kursi, dan film yang ingin ditonton. Berikut merupakan kode program bioskop.

|  |
| --- |
| #include <stdio.h>  #include <stdlib.h>  #include <string.h>  #include <time.h>  #include <windows.h>  //variabel global untuk fungsi pemesanan  float jam1[6]= {0,12.00, 14.00, 16.00, 20.00, 22.00};  float jam2[6]= {0,12.30, 14.30, 16.30, 20.30, 22.30};  float jamtayang;  int hari, film, h, i, angka\_kursi[50];  char seat[10] = {'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'};  char huruf\_kursi;  void menu();  void login\_admin();  void menu\_admin();  void riwayat\_transaksi();  void data\_customer();  void regis\_customer();  void login\_customer();  void menu\_customer();  void genreFilm();  void horor();  void komedi();  void romantis();  void pemesanan();  void pembayaran();  void keluar();  void close(){  fflush(stdin);  printf("\n\tTekan Enter untuk Melanjutkan...");  getchar();  system("cls");  }  struct customer{  char nama\_cust[50];  int pin\_cust;  };  struct customer registrasi;  struct customer login;  typedef struct{  int id;  char title[50];  char genre[20];  int stock;  float weekday\_price;  float weekend\_price;  }Movie;  void main(){  system("cls");  printf("\n\t==================================================");  printf("\n\t|| Nama Kelompok ||");  printf("\n\t==================================================");  printf("\n\t|| 1. Putu Satria Arya Putra (2305551122) ||");  printf("\n\t|| 2. Anthony Wisnu Jati (2305551138) ||");  printf("\n\t|| 3. Azka Ilham Ramadhani (2305551171) ||");  printf("\n\t==================================================");  printf("\n\tEnter untuk melanjutkan...");  while (getchar() != '\n');  menu();  }  //menu awal program  void menu(){  int pil;  system("cls");  printf("\n\t==================================================");  printf("\n\t|| Pilihan Masuk ||");  printf("\n\t==================================================");  printf("\n\t|| 1. Masuk Sebagai Admin ||");  printf("\n\t|| 2. Menu Pelanggan ||");  printf("\n\t|| 3. Exit ||");  printf("\n\t==================================================");  printf("\n\t Masukkan Pilihan Anda: ");  scanf("%d", &pil);  if(pil==1){  login\_admin();  }  else if(pil==2){  menu\_customer();  }  else if(pil==3){  keluar();  }  else{  printf("Error\n");  }  }  //tampilan log in admin  void login\_admin(){  char username [15];  char password [15];  printf("\nMasukkan username: ");  scanf("%s", &username);  printf("\nMasukkan password: ");  fflush(stdin);  scanf("%s", &password);  if ((strcmp(username, "admin") == 0) && (strcmp(password, "123") ==0)){  getchar();  printf("\n\n\tTekan Enter untuk melanjutkan...");  getchar();  system ("cls");  menu\_admin();  }else {  printf("\nPassword atau Username Salah!\n");  getchar();  printf("\nTEKAN ENTER UNTUK KEMBALI");  getchar();  system ("cls");  menu();  }  }  //tampilan menu admin  void menu\_admin(){  int pil;  while (pil!=4){  system("cls");  printf("\n\t==============================================================");  printf("\n\t|| Tampilan Menu Admin ||");  printf("\n\t||==========================================================||");  printf("\n\t|| [1] Menampilkan riwayat transaksi ||");  printf("\n\t||----------------------------------------------------------||");  printf("\n\t|| [2] Log out ||");  printf("\n\t==============================================================");  printf("\n\tMasukkan pilihan Anda [1][2]: ");  scanf("%d", &pil);  switch(pil){  case 1:  {  riwayat\_transaksi();  break;  }  case 2:{  keluar();  break;  }  default:  {  printf("Menu tidak tersedia!");  break;  }  }  }  }  void menu\_customer(){//Menampilkan pilihan menu awal yang dapat dipilih customer untuk masuk ke program  int menu;  printf("\n\t==============================================================");  printf("\n\t|| SELAMAT DATANG ||");  printf("\n\t||==========================================================||");  printf("\n\t|| [1] Registrasi ||");  printf("\n\t||----------------------------------------------------------||");  printf("\n\t|| [2] Login ||");  printf("\n\t||----------------------------------------------------------||");  printf("\n\t|| [3] Exit ||");  printf("\n\t==============================================================");  printf("\n\tMasukkan Pilihan Anda : ");  while(scanf("%d", &menu)==0 || menu < 1 || menu > 3){  printf("\t--------------------------------------------------------------");  printf("\n\tAngka yang Anda inputkan salah!");  printf("\n\tMohon inputkan pilihan yang benar...");  printf("\n\t--------------------------------------------------------------");  printf("\n\tMasukkan pilihan Anda: ");  while((getchar())!='\n');  }  close();  if (menu==1){  regis\_customer(); // menuju bagian registrasi (apabila customer belum terdaftar)  }else if(menu==2){  login\_customer(); // menuju bagian login customer(apabila customer sudah pernah mendaftarkan PIN dan usernamenya)  }else{  main(); //kembali ke menu awal  }  }  void regis\_customer(){  int i;  struct customer cust01;  FILE \*regis = fopen("DataCustomer.txt", "a+");  printf("\n\t=========================::::::::::::::::=====================\n");  printf("\n\t|| -- MENU REGISTRASI -- ||\n");  printf("\n\t=========================::::::::::::::::=====================\n");  printf("\n\tMasukkan username : ");  scanf("%[^\n]", registrasi.nama\_cust);  fflush(stdin);  //Mengecek data customer dalam file DataCustomer.txt  while (!feof(regis)){  //membaca data customer dalam file DataCustomer.txt  fscanf(regis, "%[^#]#%d\n", &cust01.nama\_cust, &cust01.pin\_cust);  fflush(stdin);  //mengecek apakah username yang diinputkan sudah pernah terdaftar sebelumnya. Apabila sudah, customer akan diminta menginputkan username baru.  if (strcmp(cust01.nama\_cust, registrasi.nama\_cust)==0){  printf("\n\t -- username sudah terdaftar. Coba gunakan username lainnya -- ");  printf("\n\t==============================================================\n");  fclose(regis);  close();  regis\_customer();  break;  }  }  //ketika username yang diinputkan belum pernah terdaftar, maka program akan lanjut meminta inputan PIN  fflush(stdin);  printf("\n\tMasukkan PIN : ");  while(scanf("%d", &registrasi.pin\_cust)==0 || registrasi.pin\_cust < 0){  printf("\t--------------------------------------------------------------");  printf("\n\tAngka yang Anda inputkan salah!");  printf("\n\tMohon inputkan angka yang benar...");  printf("\n\t--------------------------------------------------------------");  printf("\n\tMasukkan PIN : ");  while((getchar())!='\n');  }  fprintf(regis, "%s#%d\n", registrasi.nama\_cust, registrasi.pin\_cust);  fclose(regis);  printf("\n\t\tRegistrasi dengan username %s berhasil!", registrasi.nama\_cust);  printf("\n\t --------------------------------------------------------------\n");  close();  login\_customer(); //setelah registrasi berhasil, customer akan diminta untuk login  }  void login\_customer(){  int i;  int kode;  struct customer cust01;  FILE \*regis = fopen("DataCustomer.txt", "a+");  printf("\n\t=========================::::::::::::::::=====================\n");  printf("\n\t|| -- MENU LOGIN -- ||\n");  printf("\n\t=========================::::::::::::::::=====================\n");  printf("\n\tMasukkan PIN : ");  while(scanf("%d", &kode)==0 || kode < 0){  printf("\t--------------------------------------------------------------");  printf("\n\tAngka yang Anda inputkan salah!");  printf("\n\tMohon inputkan angka yang benar...");  printf("\n\t--------------------------------------------------------------");  printf("\n\tMasukkan PIN : ");  while((getchar())!='\n');  }  // mengecek data customer dalam file DataCustomer.txt  while (!feof(regis)){  //membaca data customer dalam file DataCustomer.txt  fscanf(regis, "%[^#]#%d\n", &cust01.nama\_cust, &cust01.pin\_cust);  fflush(stdin);  //mengecek apakah PIN yang diinputkan telah terdaftar  if (cust01.pin\_cust == kode){  // jika PIN telah terdaftar maka akan ditampilkan username customer  printf("\n\t==============================================================");  printf("\n\t\t\t\t LOGIN BERHASIL");  printf("\n\t--------------------------------------------------------------");  printf("\n\tHello, %s !", cust01.nama\_cust);  printf("\n\t==============================================================");  strcpy(login.nama\_cust, cust01.nama\_cust);  fclose(regis);  close();  menu\_pesan();  break;  }  }  fclose(regis);  //Jika PIN yang diinputkan belum terdaftar, program akan kembali ke fungsi menu\_customer()  printf("\n\t------------------- PIN tidak Terdaftar ---------------------");  printf("\n\t Silahkan Periksa Ulang PIN atau Registrasi Terlebih Dahulu!");  printf("\n\t==============================================================\n");  close();  menu\_pesan();  }  void menu\_pesan(){  int menu;  system("cls");  printf("\n\t===============================================");  printf("\n\t|| SELAMAT DATANG DIBIOSKOP TI ||");  printf("\n\t===============================================");  printf("\n\t|| Tampilan Menu ||");  printf("\n\t===============================================");  printf("\n\t|| 1. Menu Pemesanan Tiket ||");  printf("\n\t|| 2. Exit ||");  printf("\n\t===============================================");  printf("\n\t Masukkan Pilihan Anda: ");  scanf("%d", &menu);  if(menu==1){  genreFilm();  }  else if(menu==2){  keluar();  }  else{  printf("Error\n");  }  }  void genreFilm(){  int pil;  system("cls");  printf("\n\t\t=========================");  printf("\n\t\t|| Genre Film ||");  printf("\n\t\t=========================");  printf("\n\t\t|| 1. Horor ||");  printf("\n\t\t|| 2. Komedi ||");  printf("\n\t\t|| 3. Romantis ||");  printf("\n\t\t=========================");  printf("\n\t\t Masukkan Pilihan Genre Film Anda: ");  scanf("%d", &pil);  if(pil==1){  horor();  }  else if(pil==2){  komedi();  }  else if(pil==3){  romantis();  }  else{  printf("\n\tMasukkan Pilihan Yang Benar");  }  }  void horor(){  Movie movies[] = {  {1, "Conjuring", "Horor", 200, 80000, 90000},  {2, "Annabelle Creation", "Horor", 200, 85000, 100000},  {3, "The Ring", "Horor", 200, 90000, 110000},  };  pemesanan(movies, 3);  }  void komedi(){  Movie movies[] = {  {4, "Friends", "Komedi", 200, 60000, 65000},  {5, "Modern Family", "Komedi", 200, 70000, 80000},  {6, "Curb Your Enthusiasm", "Komedi", 200, 65000, 75000},  };  pemesanan(movies, 3);  }  void romantis(){  Movie movies[] = {  {7, "Pride and Prejudice", "Romantis", 200, 70000, 80000},  {8, "Love Actually", "Romantis", 200, 75000, 85000},  {9, "When Harry Met Sally", "Romantis", 200, 80000, 90000},  };  pemesanan(movies, 3);  }  void pemesanan(Movie movies[], int jumlah\_movie) {  int id\_movie, jumlah\_tiket, hari, i;  int cari = 0;  double price;  struct tm \*local;  time\_t now;  now = time(NULL);  local = localtime(&now);  system("cls");  printf("\n\t=====================================================================================================");  printf("\n\t|| Daftar Film yang Tayang ||");  printf("\n\t=====================================================================================================");  printf("\n\t ID \t| Title \t\t| Genre \t| Stock \t| Price Weekday | Price Weekend");  printf("\n\t-----------------------------------------------------------------------------------------------------");  for (i = 0; i < jumlah\_movie; i++) {  printf("\n\t %-2d \t| %-20s \t| %-10s \t| %-7d \t| %-12.2lf \t| %-12.2lf", movies[i].id, movies[i].title, movies[i].genre, movies[i].stock, movies[i].weekday\_price, movies[i].weekend\_price);  }  printf("\n\t Silahkan pilih ID film yang akan dipesan: ");  scanf("%d", &id\_movie);  for (i = 0; i < jumlah\_movie; i++) {  if (movies[i].id == id\_movie) {  cari = 1;  break;  }  }  hari = local->tm\_wday;  if(hari==0){  printf("\n\t Hari minggu");  }else if(hari==1){  printf("\n\t Hari senin");  }else if(hari==2){  printf("\n\t Hari selasa");  }else if(hari==3){  printf("\n\t Hari rabu");  }else if(hari==4){  printf("\n\t Hari kamis");  }else if(hari==5){  printf("\n\t Hari jumat");  }else {  printf("\n\t Hari sabtu");  }  printf(" adalah hari ke %d/%d/%d", local->tm\_mday, local->tm\_mon+1, local->tm\_year+1900);  if (cari == 1) {  switch (hari) {  case 0:  case 6:  price = movies[i].weekend\_price;  break;  default:  price = movies[i].weekday\_price;  break;  }  printf("\n\t Film yang Anda pilih adalah '%s'. Harga tiketnya adalah Rp %.2lf", movies[i].title, price);  printf("\n\t Berapa jumlah tiket yang ingin Anda pesan: ");  scanf("%d", &jumlah\_tiket);  if (price = movies[i].weekday\_price)  {  printf("\n\t%.2f \t%.2f \t%.2f \t%.2f \t%.2f ", jam1[1],jam1[2],jam1[3],jam1[4],jam1[5]);  }else if (price = movies[i].weekend\_price)  {  printf("\n\t%.2f \t%.2f \t%.2f \t%.2f \t%.2f ", jam2[1],jam2[2],jam2[3],jam2[4],jam2[5]);  }  printf("\n\t Silahkan Masukkan Jam tayang : ");  scanf("%f", &jamtayang);  printf("\n\tPilih Seat Yang Tersedia!");  printf("\n\t========================================================================");  printf("\n\t ``````````````````````");  printf("\n\t || ||");  printf("\n\t || SCREEN ||");  printf("\n\t || ||");  printf("\n\t || ||");  printf("\n\t ``````````````````````");  printf("\n\t========================================================================\n");  for(int i=0; i<10; i++){  printf("\t");  for(h=1; h<=20; h++){  printf("%c%d ", seat[i], h);//menampilkan daftar pilihan row dan seat  }  printf("\n\t------------------------------------------------------------------------\n");  }  printf("\t========================================================================");  printf("\n\tPilih Seat (A-J) : ");  fflush(stdin);  scanf("%c", &huruf\_kursi);  printf("\n\tAnda memilih seat %c, Selanjutnya silahkan pilih nomor kursi!", huruf\_kursi);  for(h=0; h<jumlah\_tiket; h++){  cekkursi:  printf("\n\tPilih Nomor kursi ke-%d : ", h+1);  while(scanf("%d", &angka\_kursi[h])==0 || angka\_kursi[h] < 1 || angka\_kursi[h] > 20){  printf("\t------------------------------------------------------------------------");  printf("\n\tNomor Kursi Tidak Tersedia!");  printf("\n\tMohon masukkan ulang pilihan");  printf("\n\t------------------------------------------------------------------------");  printf("\n\tMasukkan Pilihan Anda : ");  while((getchar())!='\n');  }  for(int i=0; i<h; i++){  if(angka\_kursi[h]==angka\_kursi[i]){  printf("\t--------------------------------------------------------------");  printf("\n\tMaaf, Kursi %c-%d telah terisi, silahkan pilih kembali",huruf\_kursi, angka\_kursi[i]);  printf("\n\t--------------------------------------------------------------");  goto cekkursi;  }  }  }  if (jumlah\_tiket > movies[i].stock) {  printf("\n\t Maaf, stock tiket yang tersedia adalah %d", movies[i].stock);  } else {  movies[i].stock -= jumlah\_tiket;  pembayaran(movies[i], jumlah\_tiket, price);  }  } else {  printf("\n\t Maaf, ID film yang Anda masukkan tidak valid! ");  }  }  void pembayaran(Movie movie, int jumlah\_tiket, double price) {  double total\_harga = price \* jumlah\_tiket;  char cetak;  struct tm \*local;  time\_t now;  now = time(NULL);  local = localtime(&now);  double uang\_bayar;  system("cls");  printf("\n\t=============================================");  printf("\n\t|| PEMBAYARAN TIKET ||");  printf("\n\t=============================================");  printf("\n\t Film yang anda pesan: %s", movie.title);  printf("\n\t Jam tayang : %.2f", jamtayang);  printf("\n\t Jumlah tiket yang anda pesan: %d", jumlah\_tiket);  printf("\n\t---------------------------------------------");  printf("\n\t Total harga pembelian tiket adalah Rp %.2lf", total\_harga);  printf("\n\t---------------------------------------------");  do  {  printf("\n\t Silahkan masukkan jumlah uang yang akan Anda bayarkan: ");  scanf("%lf", &uang\_bayar);  if (uang\_bayar < total\_harga) {  printf("\n\t Maaf, uang yang Anda masukkan tidak cukup untuk membayar tiket! ");  } else {  double kembalian = uang\_bayar - total\_harga;  printf("\n\t Terima kasih atas pemesanan Anda! Tiket yang Anda pesan telah berhasil dipesan. ");  printf("\n\t Uang kembalian Anda adalah Rp %.2lf", kembalian);  }  } while (uang\_bayar < total\_harga);  printf("\n\t=========================================================================\n");  printf("\t| NO |\t\tFILM\t\t|\tJam Tayang\t| Harga |\n");  printf("\t=========================================================================");  for (int i=0; i<jumlah\_tiket; i++){  printf("\n\t| %d |\t%s\t | \t%.2f |\t%.2f|", i+1, movie.title, jamtayang, price);  }  printf("\n\t=========================================================================\n");  printf("\t| Total Harga : %.2f \t\t\t\t\t\t|\n",total\_harga);  printf("\t=========================================================================");  printf("\n\tSimpan Tiket (Y/N)? ");  fflush(stdin);  scanf("%c", &cetak );  if(cetak=='y' | cetak=='Y'){  for(int i=0; i<jumlah\_tiket; i++){  // Untuk menyimpan di DataPesanan.txt  FILE \*insert = fopen("DataPesanan.txt", "a");  fprintf(insert,"\n\t=================================");  fprintf(insert,"\n\t|\t Bioskop TI\t\t|");  fprintf(insert,"\n\t=================================");  fprintf(insert,"\n\t|\t%s\t|", movie.title);  fprintf(insert,"\n\t| date : %d/%d/%d \t\t|",local->tm\_mday, local->tm\_mon+1, local->tm\_year+1900);  fprintf(insert,"\n\t| time : %.2f\t\t\t|",jamtayang);  fprintf(insert,"\n\t| row : %c seat : %d\t\t|", huruf\_kursi, angka\_kursi[i]);  fprintf(insert,"\n\t| price : %.3f \t\t|", price);  fprintf(insert,"\n\t=================================");  fclose(insert);  // Untuk menyimpan di RiwayatTransaksi.txt  FILE \*file = fopen("RiwayatTransaksi.txt", "a");  fprintf(file,"\n\t=================================");  fprintf(file,"\n\t|\t Bioskop TI\t\t|");  fprintf(file,"\n\t=================================");  fprintf(file,"\n\t|\t%s\t|", movie.title);  fprintf(file,"\n\t| date : %d/%d/%d \t\t|",local->tm\_mday, local->tm\_mon+1, local->tm\_year+1900);  fprintf(file,"\n\t| time : %.2f\t\t\t|",jamtayang);  fprintf(file,"\n\t| row : %c seat : %d\t\t|", huruf\_kursi, angka\_kursi[i]);  fprintf(file,"\n\t| price : %.3f \t\t|", price);  fprintf(file,"\n\t=================================");  fclose(file);  printf("\n\t=================================");  printf("\n\t|\t Bioskop TI\t\t|");  printf("\n\t=================================");  printf("\n\t|\t%s\t|", movie.title);  printf("\n\t| date : %d/%d/%d \t\t|",local->tm\_mday, local->tm\_mon+1, local->tm\_year+1900);  printf("\n\t| time : %.2f\t\t\t|",jamtayang);  printf("\n\t| row : %c seat : %d\t\t|", huruf\_kursi, angka\_kursi[i]);  printf("\n\t| price : %.3f \t\t|", price);  printf("\n\t=================================");  }  }  }  void riwayat\_transaksi(){  char pilih;  system("cls");  char buff[255];  FILE \*header;  printf("\n\t==============================================================\n");  printf("\n\t|| Riwayat Transaksi ||");  printf("\n\t==============================================================");  if ((header = fopen("RiwayatTransaksi.txt","r")) == NULL){  printf("\n\tError: File tidak ada!");  }  while(fgets(buff, sizeof(buff), header)){  printf("%s", buff);  }  fclose(header);  printf("\nApakah Mau kembali ke menu admin ? (Y/N) : ");  fflush(stdin);  scanf("%c", &pilih);  if(pilih=="Y"||pilih=="y"){  menu\_admin();  }else{  keluar();  }  }  void keluar() {  printf("\n\t Terima Kasih Telah Menggunakan Sistem Pemesanan Tiket di Bioskop TI ");  printf("\n\t Selamat Datang Kembali! ");  getchar();  exit(0);  } |

**Kode Program 2. 2** Program Bioskop

Kode program 2.2 merupakan program bioskop yang sudah dibuat dengan menggunakan Bahasa C. program diatas juga sudah sesuai dengan *flowchart* dan *pseudocode* sebelumnya. Program tersebut dibuat Bersama-sama oleh anggota kelompok kami dengan cara membagi tugas untuk masing-masing anggota kelompok.

# BAB III PENUTUP

Bab ini merupakan bagian akhir dari laporan yang dibuat oleh penulis laporan untuk mata kuliah Algoritma dan Pemrograman. Bab ini memuat kesimpulan dari isi materi laporan serta kritik dan saran dari berbagai pihak seperti teman-teman, dan dosen pengajar algoritma dan pemrograman.

## 3.1 Simpulan

Pada pembuatan laporan Algoritma Pemrograman, adapun simpulan yang dapat diambil yang diuraikan sebagai berikut. Pemrograman Dasar, Fungsi dan Prosedur berisi berbagai macam pertanyaan tentang program bioskop. Program di atas dirancang terlebih dahulu dengan konsep flowchart dan pseudocode kemudian diimplementasikan menggunakan bahasa pemrograman C/C++.

## 3.2 Saran

Pada pembuatan laporan Algoritma Pemrograman ini adapun saran yang dapat diberikan di dalamnya yakni:

1. Pertama, perlu diperhatikan, bahwa dalam membuat sebuah program haruslah memperhatikan setiap tahapan dalam proses pembuatannya dan juga mengerti bagaimana alur dari program yang kita buat itu berjalan.
2. Kedua, penyusun mengharapkan semua mahasiswa menjadi diri sendiri tanpa harus mementingkan perkataan orang lain dan juga membandingkan kemampuan diri sendiri dengan orang lain tanpa tujuan ingin mengembangkan diri, serta menjadi kreatif dan dapat merancang berbagai program sendiri dengan lebih baik.