

LAPORAN FINAL PROJECT
STRUKTUR DATA C081
SISTEM PEMBAYARAN PARKIR

Dosen Pengampu :
Fawwaz Ali Akbar, S.Kom., M.Kom.



DISUSUN OLEH :

| | |
|-------------------------------------|--------------------|
| Desya Ristya Putri | 20081010240 |
| Faris Munir Mahdi | 21081010064 |
| Nauval Maulana Rizky Irawan | 21081010066 |
| Achmad Robby Sabilul Firdaus | 21081010065 |

PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"
JAWA TIMUR

2022

Sistem Pembayaran Parkir

Latar Belakang

Akhir-akhir ini, banyak bermunculan perusahaan pembuat sistem parkir. Hal tersebut disebabkan karena mayoritas tempat telah menerapkan biaya untuk siapapun yang parkir di wilayahnya, mulai dari pasar, pertokoan, tempat perbelanjaan, mall, dan lain sebagainya. Sistem parkir yang masih manual (tanpa aplikasi) cukup rentan akan tindak kejahatan. Tindakan curang juga terkadang dilakukan oleh pengelola parkir, misalnya menaikkan biaya parkir diluar ketentuan yang ada.

Dari kasus tersebut, muncul sebuah ide dari penulis untuk membuat sebuah aplikasi yang digunakan untuk mengelola parkir sehingga Final Project ini berjudul “Sistem Pembayaran Parkir”. Aplikasi dibangun menggunakan bahasa pemrograman C dengan menerapkan dua struktur data, yaitu **Queue** dan **Single Linked List**. Dalam aplikasi ini juga terdapat fitur prioritas yang dapat memberikan layanan khusus berupa menyela antrean bagi kendaraan pelanggan yang sesuai dengan kriteria. Struktur data Queue digunakan untuk menyimpan data antrean kendaraan dengan metode *First in First Out* (FIFO) sedangkan Single Linked List digunakan untuk menyisipkan atau menambahkan data kendaraan yang memiliki prioritas untuk menyela antrean.

Tujuan

1. Membuat program yang dapat membantu dalam pengelolaan parkir
2. Membuat program yang fungsional
3. Mengimplementasikan struktur data yang telah diajarkan
4. Menyelesaikan Final Project mata kuliah Struktur Data

Studi Kasus

Di sebuah pertokoan, sistem untuk parkir kendaraan sangat diperlukan. Selain untuk keamanan, parkir juga dapat menjadi satu diantara sumber pendapatan pertokoan. Pada pertokoan X menerapkan sistem parkir dimana data yang dikelola adalah sebagai berikut.

1. Nomor Plat Kendaraan
2. Warna Kendaraan
3. Durasi Parkir (Dalam Menit)
4. Golongan Kendaraan (R2/R4)

Selain itu biaya parkir kendaraan memiliki ketentuan sebagai berikut.

1. Setiap tanggal ganjil dan no plat kendaraan ganjil, maka pembayaran parkir diskon 50%, berlaku juga untuk tanggal dan no plat genap.
2. Pelanggan mendapat prioritas berupa dapat menyela antrean ke antrean paling depan apabila warna kendaraan sesuai dengan warna yang ditentukan pada periode setiap harinya. Kendaraan yang mendapat prioritas merupakan kendaraan dengan warna dan hari yang sesuai dengan ketentuan berikut.
 - a. Senin: hitam
 - b. Selasa: putih
 - c. Rabu: Selain hitam dan putih
 - d. Kamis: hitam
 - e. Jumat: putih
 - f. Sabtu: Selain hitam dan putih
 - g. Minggu: hitam
3. Parkir dengan durasi ≤ 5 menit tidak dikenakan biaya (gratis).
4. Parkir pada 60 menit pertama dikenakan biaya:
 - a. R2: 5000
 - b. R4: 10000Untuk durasi selanjutnya, dikenakan biaya 1000 setiap 60 menit berikutnya.

Fitur yang Ditawarkan

1. Menambahkan data kendaraan yang parkir (menggunakan queue untuk kendaraan normal, menggunakan single linked list untuk kendaraan dengan prioritas)
2. Menghitung total biaya parkir dan menghapus datanya ketika sudah selesai transaksi
3. Menampilkan seluruh data kendaraan yang parkir
4. Search data kendaraan

Link Github

<https://github.com/desyaristya/sistem-pembayaran-parkir>

Link Video Presentasi Final Project

<https://drive.google.com/drive/u/3/folders/1ZZHqlayARHnCB-MyyQIB2gxYhvLAX6FR>

Terdapat dua video presentasi, yaitu:

1. Video penjelasan project dan penjelasan kode program
2. Video demo program

Source Code

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <time.h>
4  #include <conio.h>
5  #include <string.h>
6  #include <math.h>
7  #define MAX 30
8
```

Kode di atas merupakan library yang digunakan pada program. Library time.h digunakan untuk mengambil data waktu lokal device sedangkan library math.h digunakan untuk menggunakan fungsi ceil() guna melakukan pembulatan durasi parkir.

```
9  struct parkir{
10     char hurufDepanPlat[MAX];
11     int noPlat;
12     char hurufBelakangPlat[MAX];
13     char warnaKendaraan[MAX];
14     int durasi;
15     char golongan[MAX];
16     struct parkir *next;
17 }*front, *frontNode, *rear, *temp, *p;
18
19 int count = 0;
20
```

Kode di atas merupakan pembuatan struct yang digunakan sebagai penampung data kendaraan yang parkir. Selain itu, juga terdapat inisialisasi variable count yang digunakan untuk menghitung banyaknya node.

```
21 void menu()
22 {
23     printf("-----\n");
24     printf("          SISTEM PEMBAYARAN PARKIR KENDARAAN PERTOKOAN X\n");
25     printf("-----\n");
26     printf("1. Memasukkan data kendaraan baru ke dalam antrean pembayaran\n");
27     //terdapat dua pilihan, yaitu enqueue pada queue dan add front pada single linkedlist
28     printf("2. Hitung biaya parkir, lalu hapus data ketika pembayaran selesai\n");
29     //dequeue pada queue
30     printf("3. Tampilkan seluruh data antrean kendaraan\n");
31     printf("4. Cari data kendaraan berdasarkan No Plat-nya\n");
32     printf("5. Exit\n");
33     printf("-----\n");
34 }
```

Function menu di atas digunakan untuk menampilkan menu awal pada program, dimana terdapat pilihan untuk memasukkan data kendaraan baru, menghitung biaya parkir (hapus

kendaraan), menampilkan keseluruhan antrean, dan mencari data kendaraan berdasarkan plat nomornya.

```
34 int currentTanggal(){
35     time_t rawtime = time(NULL);
36     struct tm *ptm = localtime(&rawtime);
37     int tanggal = ptm->tm_mday;
38     return tanggal;
39 }
40
```

Function currentTanggal di atas digunakan untuk membaca tanggal saat ini (realtime). Hal ini dilakukan untuk pengecekan tanggal saat ini dengan nomor plat kendaraan pada diskon ganjil genap.

```
41 int currentHari(){
42     time_t rawtime = time(NULL);
43     struct tm *ptm = localtime(&rawtime);
44     int hari = ptm->tm_wday;
45     return hari;
46 }
47
```

Function currentHari di atas digunakan untuk membaca hari saat ini (realtime). Return dari function ini berupa nilai integer dari 0-6 yang mewakili hari Sunday-Saturday. Pembacaan hari saat ini pada function di atas dipakai ketika pengecekan prioritas kendaraan, apakah warna kendaraan sesuai dengan warna pada ketentuan hari ini atau tidak.

```
48 int compareString(char value1[], char value2[]){
49     int result;
50     result = strcmp(value1, value2);
51     return result;
52 }
53
```

Function compareString di atas digunakan untuk membandingkan dua buah string dengan nilai return berupa 0 yang berarti string tersebut memiliki nilai yang sama ataupun selain 0 yang berarti string tersebut memiliki nilai yang berbeda.

```
54 void createQueue(){
55     front = NULL;
56     rear = NULL;
57 }
58
```

Function createQueue di atas digunakan untuk mendefinisikan awalan dan akhiran dari antrean, yaitu NULL (belum terisi data, dan masih antrean kosong).

```
59 void enqueue(char newPlatDepan[], int newPlatNo, char newPlatBelakang[], char
60     newWarna[], int newDur, char newGol[]){
61     fflush(stdin);
```

```

62     if (rear == NULL){
63         rear = (struct parkir *)malloc(1*sizeof(struct parkir));
64         rear->next = NULL;
65         strcpy(rear->hurufDepanPlat, newPlatDepan);
66         rear->noPlat = newPlatNo;
67         strcpy(rear->hurufBelakangPlat, newPlatBelakang);
68         strcpy(rear->warnaKendaraan, newWarna);
69         rear->durasi = newDur;
70         strcpy(rear->golongan, newGol);
71         front = rear;
72     }
73     else{
74         temp=(struct parkir *)malloc(1*sizeof(struct parkir));
75         rear->next = temp;
76         strcpy(temp->hurufDepanPlat, newPlatDepan);
77         temp->noPlat = newPlatNo;
78         strcpy(temp->hurufBelakangPlat, newPlatBelakang);
79         strcpy(temp->warnaKendaraan, newWarna);
80         temp->durasi = newDur;
81         strcpy(temp->golongan, newGol);
82         temp->next = NULL;
83         rear = temp;
84     }
85     count++;
86 }
87

```

Function enqueue di atas digunakan untuk menambahkan fungsi untuk data yang akan ditambahkan pada antrean yang terdiri dari plat nomor kendaraan, warna kendaraan, durasi, dan golongan kendaraan. Sesuai dengan konsep queue, penambahan data baru akan diletakan di paling belakang. If dengan kondisi rear == NULL digunakan apabila antrean masih kosong sedangkan else digunakan apabila antrean sudah terisi kendaraan.

```

88 void addDataPrioritas(char newPlatDepan[], int newPlatNo, char
89 newPlatBelakang[], char newWarna[], int newDur, char newGol[]){
90     fflush(stdin);
91     if (temp != NULL){
92         temp=(struct parkir *)malloc(1*sizeof(struct parkir));
93         strcpy(temp->hurufDepanPlat, newPlatDepan);
94         temp->noPlat = newPlatNo;
95         strcpy(temp->hurufBelakangPlat, newPlatBelakang);
96         strcpy(temp->warnaKendaraan, newWarna);
97         temp->durasi = newDur;
98         strcpy(temp->golongan, newGol);
99         temp->next = front;
100        front = temp;
101    }
102    count++;
103 }
104

```

Function addDataPrioritas di atas digunakan untuk menambahkan fungsi apabila ada kendaraan prioritas yang ditambahkan. Kendaraan yang memiliki prioritas dapat menyela

antrean di paling depan sehingga penambahan data ini menggunakan konsep single linked list, yaitu add data pada bagian depan.

```
105 int diskonGanjilGenap(int platNo, int biaya){
106     int tanggalCur = currentTanggal();
107
108     //tanggal dan plat nomor sama-sama ganjil atau sama-sama genap
109     if((tanggalCur % 2 == 0 && platNo % 2 == 0) || (tanggalCur % 2 != 0 &&
    platNo % 2 != 0)){
110         biaya = biaya * 0.5;
111     }
112
113     //tanggal dan plat nomor tidak sama-sama ganjil atau tidak sama-sama
    genap
114     if((tanggalCur % 2 != 0 && platNo % 2 == 0) || (tanggalCur % 2 == 0 &&
    platNo % 2 != 0)){
115         biaya = biaya;
116     }
117
118     return biaya;
119 }
120
```

Function `diskonGanjilGenap` di atas digunakan untuk pengecekan apakah kendaraan mendapat diskon biaya parkir atau tidak. Apabila plat nomor kendaraan ganjil dan tanggal ganjil ataupun plat nomor kendaraan genap dan tanggal genap, maka kendaraan tersebut akan mendapatkan diskon sebanyak 50% dari total harga parkir.

```
121 int biayaParkir(int durasiParkir, char golKendaraan[], int platNo){
122     int biaya, biayaSetelahCekDiskon;
123     double pembulatanDurasi;
124     if(durasiParkir <= 5){
125         biaya = 0;
126     }
127     else if(durasiParkir > 5 && durasiParkir <= 60){
128         if((compareString(golKendaraan, "R2")) == 0){
129             biaya = 5000;
130         }
131         if((compareString(golKendaraan, "R4")) == 0){
132             biaya = 10000;
133         }
134     }
135     else if(durasiParkir > 60){
136         pembulatanDurasi = ceil(durasiParkir/60.0);
137         if((compareString(golKendaraan, "R2")) == 0){
138             biaya = 5000 + ((pembulatanDurasi-1)*1000);
139         }
140         if((compareString(golKendaraan, "R4")) == 0){
141             biaya = 10000 + ((pembulatanDurasi-1)*1000);
142         }
143     }
144
145     biayaSetelahCekDiskon = diskonGanjilGenap(platNo, biaya);
146     return biayaSetelahCekDiskon;
147 }
```

Function biayaParkir di atas digunakan untuk menghitung total biaya parkir kendaraan.

- Jika kendaraan parkir kurang dari atau sama dengan 5 menit, maka biaya parkir gratis.
- Jika kendaraan parkir kurang dari atau sama dengan 60 menit, maka pengemudi diharuskan membayar sebanyak Rp 5000 untuk kendaraan beroda dua dan Rp 10000 untuk kendaraan beroda empat.
- Jika kendaraan parkir lebih dari 60 menit, maka pengemudi diharuskan membayar penambahan biaya parkir sebanyak Rp 1000 untuk setiap jam selanjutnya. Untuk kendaraan beroda dua akan membayar Rp 5000 + tambahan biaya pada jam berikutnya sedangkan untuk kendaraan beroda empat akan membayar Rp 10000 + tambahan biaya pada jam berikutnya.

Di akhir, dilakukan pengecekan apakah kendaraan yang parkir mendapat diskon ganjil genap atau tidak dengan memanggil function diskonGanjilGenap.

```

149 void dequeue(){
150     fflush(stdin);
151     frontNode = front;
152     int cash, change;
153
154
155     if (frontNode == NULL){
156         printf("\nBelum ada data yang tersedia!");
157         return;
158     }
159     else{
160         if (frontNode->next != NULL){
161             frontNode = frontNode->next;
162             printf("No plat kendaraan: %s %d %s\n", front->hurufDepanPlat,
163                 front->noPlat, front->hurufBelakangPlat);
164             printf("Biaya parkir kendaraan: %d\n", biayaParkir(front->durasi,
165                 front->golongan, front->noPlat));
166             printf("Masukkan jumlah uang tunai: ");
167             scanf("%d", &cash);
168             change = cash - biayaParkir(front->durasi, front->golongan,
169                 front->noPlat);
170             printf("Kembalian: %d", change);
171             printf("\n\nTERIMA KASIH TELAH MELAKUKAN PEMBAYARAN\n");
172             printf("Data dengan plat nomor kendaraan %s %d %s telah
173                 terhapus!", front->hurufDepanPlat, front->noPlat,
174                 front->hurufBelakangPlat);
175             free(front);
176             front = frontNode;
177         }
178         else{
179             printf("No plat kendaraan: %s %d %s\n", front->hurufDepanPlat,
180                 front->noPlat, front->hurufBelakangPlat);
181             printf("Biaya parkir kendaraan: %d\n", biayaParkir(front->durasi,
182                 front->golongan, front->noPlat));
183             printf("Masukkan jumlah uang tunai: ");

```



```

177         scanf("%d", &cash);
178         change = cash - biayaParkir(front->durasi, front->golongan,
179                                     front->noPlat);
180         printf("Kembalian: %d", change);
181         printf("\n\nTERIMA KASIH TELAH MELAKUKAN PEMBAYARAN\n");
182         printf("Data dengan plat nomor kendaraan %s %d %s telah
183             terhapus!", front->hurufDepanPlat, front->noPlat,
184             front->hurufBelakangPlat);
185         free(front);
186         front = NULL;
187         rear = NULL;
188     }
189     count--;
190 }

```

Function dequeue di atas digunakan untuk menghapus data kendaraan yang telah menyelesaikan transaksi pembayaran parkir. Pengemudi yang ingin keluar dari lahan parkir dapat melihat berapa biaya parkir yang harus dibayar beserta jumlah kembalian apabila uang yang dikeluarkan pengemudi lebih besar dari biaya parkir. Penghapusan data kendaraan ini menggunakan konsep dequeue pada queue, yaitu menghapus node paling depan.

```

190 void printNode(struct parkir *p){
191     printf("\n|\t%s %d %s\t|\t%s\t|\t%d\t|\t%s\t|\n", p->hurufDepanPlat,
192           p->noPlat, p->hurufBelakangPlat, p->warnaKendaraan, p->durasi,
193           p->golongan);
194     printf("-----");
195 }

```

Function printNode di atas digunakan untuk menampilkan sebuah data kendaraan yang berada pada list.

```

195 void display(){
196     frontNode = front;
197
198     if ((frontNode == NULL) && (rear == NULL)){
199         printf("Belum ada data yang tersedia!");
200         return;
201     }
202
203     printf("-----");
204     printf("\n|\tPlat Nomor\t|\tWarna\t|\tDurasi\t|\tGol\t|\n");
205     printf("-----");
206     while (frontNode != rear){
207         printNode(frontNode);
208         frontNode = frontNode->next;
209     }
210     if (frontNode == rear){
211         printNode(frontNode);
212     }

```

| | |
|-----|---|
| 212 | } |
| 213 | |

Function display di atas digunakan untuk menampilkan keseluruhan data yang terdapat pada list. Pada function ini juga terdapat statement apabila terjadi kejadian tak terduga seperti apabila antrian masih kosong akan ada pesan yang menyatakan bahwa belum ada kendaraan pada antrian.

| | |
|-----|--|
| 214 | void search(char searchPlatDepan[], int searchPlatNo, char |
| | searchPlatBelakang[]) |
| 215 | { |
| 216 | frontNode = front; |
| 217 | |
| 218 | if ((frontNode == NULL) && (rear == NULL)){ |
| 219 | printf("\nBelum ada data yang tersedia!"); |
| 220 | return; |
| 221 | } |
| 222 | |
| 223 | while (frontNode != rear){ |
| 224 | if ((compareString(frontNode->hurufDepanPlat, searchPlatDepan) == 0) |
| | && (frontNode->noPlat == searchPlatNo) && |
| | (compareString(frontNode->hurufBelakangPlat, searchPlatBelakang) == |
| | 0)){ |
| 225 | printf("\n\nData kendaraan ditemukan!\n"); |
| 226 | printf("----- |
| | -----"); |
| 227 | printf("\n\tPlat Nomor\t\t\tWarna\t\t\tDurasi\t\t\tGol\t\t\n"); |
| 228 | printf("----- |
| | -----"); |
| 229 | printNode(frontNode); |
| 230 | return; |
| 231 | } |
| 232 | frontNode = frontNode->next; |
| 233 | } |
| 234 | if ((frontNode == rear) && (compareString(frontNode->hurufDepanPlat, |
| | searchPlatDepan) == 0) && (frontNode->noPlat == searchPlatNo) && |
| | (compareString(frontNode->hurufBelakangPlat, searchPlatBelakang) == 0)){ |
| 235 | printf("\nData kendaraan ditemukan!\n"); |
| 236 | printf("----- |
| | -----"); |
| 237 | printf("\n\tPlat Nomor\t\t\tWarna\t\t\tDurasi\t\t\tGol\t\t\n"); |
| 238 | printf("----- |
| | -----"); |
| 239 | printNode(frontNode); |
| 240 | return; |
| 241 | } |
| 242 | printf("\nData kendaraan dengan nomor plat %s %d %s tidak ditemukan!", |
| | searchPlatDepan, searchPlatNo, searchPlatBelakang); |
| 243 | } |
| 244 | |

Function search di atas digunakan untuk mencari data. Dengan memasukkan plat nomor kendaraan dari kendaraan yang terparkir, akan ditampilkan data-data dari kendaraan tersebut.

| | |
|-----|-------------|
| 245 | int main(){ |
|-----|-------------|

```

246 char warnaPrioritasHariIni[30];
247 int platNo, dur;
248 char platDepan[30], platBelakang[30], warna[30], gol[30];
249 char pilihMenu;
250
251 createQueue();
252
253 do{
254     system("cls");
255     menu();
256     printf("\n\nPilih menu yang tersedia: ");
257     scanf("%d", &pilihMenu);
258     switch(pilihMenu)
259     {
260         case 1:
261             system("cls");
262             printf("Masukkan plat no kendaraan (ex:
263             L<space>1234<space>LL): ");
264             scanf("%s %d %s", &platDepan, &platNo, &platBelakang);
265
266             //warna kendaraan
267             printf("Masukkan warna kendaraan (1. Hitam, 2. Putih, 3.
268             Lainnya): ");
269             scanf("%s", &warna);
270             if(compareString(warna, "1") == 0){
271                 strcpy(warna, "Hitam");
272             }
273             if(compareString(warna, "2") == 0){
274                 strcpy(warna, "Putih");
275             }
276             if(compareString(warna, "3") == 0){
277                 strcpy(warna, "Lainnya");
278             }
279
280             printf("Masukkan durasi parkir kendaraan (dalam menit): ");
281             scanf("%d", &dur);
282
283             //golongan kendaraan
284             printf("Masukkan golongan kendaraan (1. R2, 2. R4): ");
285             scanf("%s", &gol);
286             if(compareString(gol, "1") == 0){
287                 strcpy(gol, "R2");
288             }
289             if(compareString(gol, "2") == 0){
290                 strcpy(gol, "R4");
291             }
292
293             //warna prioritas berdasarkan hari
294             if(currentHari() == 0){ //sunday
295                 strcpy(warnaPrioritasHariIni, "Hitam");
296             }
297             else if(currentHari() == 1){ //monday
298                 strcpy(warnaPrioritasHariIni, "Hitam");
299             }
300             else if(currentHari() == 2){ //tuesday
301                 strcpy(warnaPrioritasHariIni, "Putih");
302             }
303             else if(currentHari() == 3){ //wednesday

```

| | |
|-----|---|
| 302 | strcpy(warnaPrioritasHariIni, "Lainnya"); |
| 303 | } |
| 304 | else if(currentHari() == 4){ //thursday |
| 305 | strcpy(warnaPrioritasHariIni, "Hitam"); |
| 306 | } |
| 307 | else if(currentHari() == 5){ //friday |
| 308 | strcpy(warnaPrioritasHariIni, "Putih"); |
| 309 | } |
| 310 | else if(currentHari() == 6){ //saturday |
| 311 | strcpy(warnaPrioritasHariIni, "Lainnya"); |
| 312 | } |
| 313 | |
| 314 | if(compareString(warna, warnaPrioritasHariIni) == 0){ |
| 315 | addDataPrioritas(platDepan, platNo, platBelakang, warna, |
| | dur, gol); |
| 316 | } |
| 317 | else{ |
| 318 | enqueue(platDepan, platNo, platBelakang, warna, dur, gol); |
| 319 | } |
| 320 | |
| 321 | system("cls"); |
| 322 | break; |
| 323 | case 2: |
| 324 | system("cls"); |
| 325 | dequeue(); |
| 326 | getch(); |
| 327 | system("cls"); |
| 328 | break; |
| 329 | case 3: |
| 330 | system("cls"); |
| 331 | display(); |
| 332 | getch(); |
| 333 | system("cls"); |
| 334 | break; |
| 335 | case 4: |
| 336 | system("cls"); |
| 337 | printf("Masukkan Plat Nomor yang Datanya Ingin Dicari"); |
| 338 | printf("\nMasukkan plat no kendaraan (ex: |
| | L<space>1234<space>LL): "); |
| 339 | scanf("%s %d %s", &platDepan, &platNo, &platBelakang); |
| 340 | search(platDepan, platNo, platBelakang); |
| 341 | getch(); |
| 342 | system("cls"); |
| 343 | break; |
| 344 | case 5: |
| 345 | exit(0); |
| 346 | break; |
| 347 | default: |
| 348 | system("cls"); |
| 349 | printf("Menu yang dipilih salah, tekan enter untuk memasukkan |
| | ulang menu!"); |
| 350 | getch(); |
| 351 | } |
| 352 | } while (pilihMenu != '5'); |
| 353 | } |

Function main berguna untuk menampilkan keseluruhan program. Pada main, dilakukan pemanggilan function-function yang diperlukan.

Hasil Eksekusi Program

1. Tampilan Awal Program

```
C:\Users\ASUS\Documents\semester 3\strukdat\final project_sistemparkir.exe

-----
                        SISTEM PEMBAYARAN PARKIR KENDARAAN PERTOKOAN X
-----
1. Memasukkan data kendaraan baru ke dalam antrean pembayaran
2. Hitung biaya parkir, lalu hapus data ketika pembayaran selesai
3. Tampilkan seluruh data antrean kendaraan
4. Cari data kendaraan berdasarkan No Plat-nya
5. Exit
-----

Pilih menu yang tersedia: █
```

Ketika program pertama dijalankan, user akan ditawarkan beberapa menu seperti yang ditampilkan pada gambar di atas.

2. Tampilan Apabila Memilih Opsi Pertama

```
C:\Users\ASUS\Documents\semester 3\strukdat\final project_sistemparkir.exe
Masukkan plat no kendaraan (ex: L<space>1234<space>LL): L 4111 YU
Masukkan warna kendaraan (1. Hitam, 2. Putih, 3. Lainnya): 2
Masukkan durasi parkir kendaraan (dalam menit): 70
Masukkan golongan kendaraan (1. R2, 2. R4): 1
```

User diminta untuk memasukkan data kendaraan yang baru masuk antrean. Data yang dimasukkan, meliputi plat nomor, warna kendaraan, durasi parkir, dan golongan kendaraan.

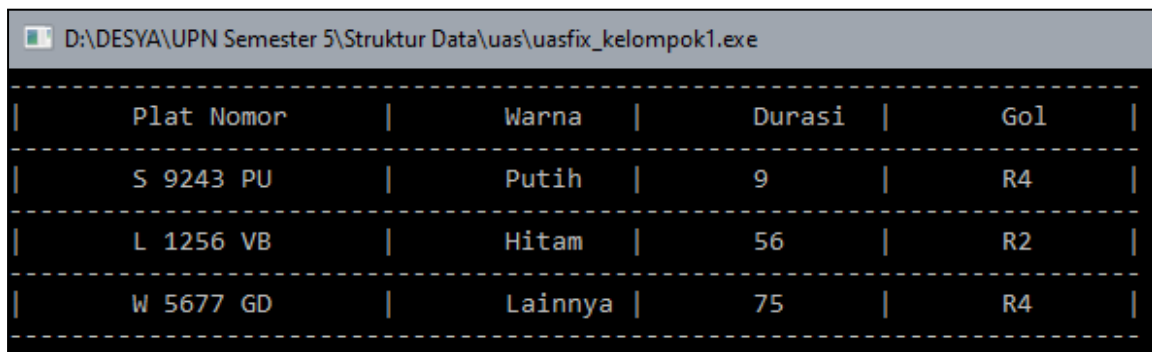
3. Tampilan Apabila Memilih Opsi Kedua

```
C:\Users\ASUS\Documents\semester 3\strukdat\final project_sistemparkir.exe
No plat kendaraan: L 1234 LL
Biaya parkir kendaraan: 5000
Masukkan jumlah uang tunai: 7000
Kembalian: 2000

TERIMA KASIH TELAH MELAKUKAN PEMBAYARAN
Data dengan plat nomor kendaraan L 1234 LL telah terhapus!
```

Opsi kedua akan menampilkan total biaya yang harus dibayarkan oleh pengemudi. Setelah selesai melakukan transaksi, data kendaraan akan dihapus.

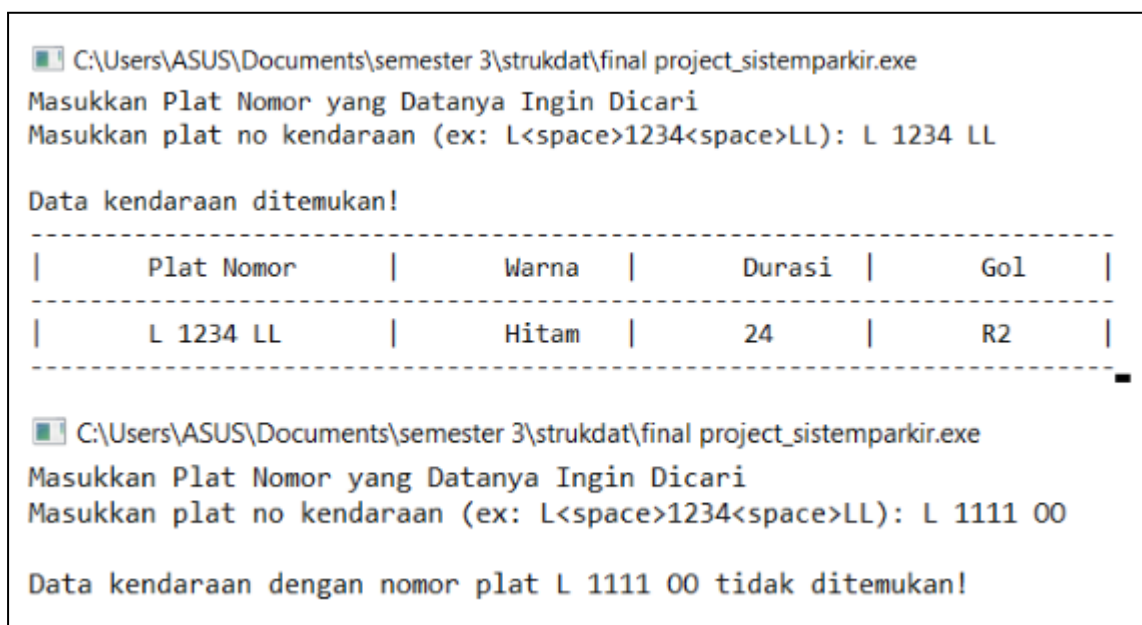
4. Tampilan Apabila Memilih Opsi Ketiga



| Plat Nomor | Warna | Durasi | Gol |
|------------|---------|--------|-----|
| S 9243 PU | Putih | 9 | R4 |
| L 1256 VB | Hitam | 56 | R2 |
| W 5677 GD | Lainnya | 75 | R4 |

Pada opsi ketiga, user akan ditampilkan seluruh data kendaraan yang terdapat pada antrean.

5. Tampilan Apabila Memilih Opsi Keempat



Gambar di atas menunjukkan tampilan ketika user memilih opsi keempat, yaitu search data kendaraan dengan memasukkan nomor platnya. Apabila data yang diinginkan tersedia, maka data kendaraan tersebut akan ditampilkan. Apabila data yang diinginkan tidak tersedia, maka ada pesan yang menyatakan bahwa data kendaraan yang dimaksud tidak terdapat pada antrean.

6. Tampilan Apabila Memilih Opsi Kelima

```
C:\Users\ASUS\Documents\semester 3\strukdat\final project_sistemparkir.exe
-----
                SISTEM PEMBAYARAN PARKIR KENDARAAN PERTOKOAN X
-----
1. Memasukkan data kendaraan baru ke dalam antrean pembayaran
2. Hitung biaya parkir, lalu hapus data ketika pembayaran selesai
3. Tampilkan seluruh data antrean kendaraan
4. Cari data kendaraan berdasarkan No Plat-nya
5. Exit
-----

Pilih menu yang tersedia: 5

-----
Process exited after 1.926 seconds with return value 0
Press any key to continue . . .
```

Apabila user memasukkan opsi kelima, maka program akan keluar atau dihentikan.

7. Tampilan Apabila Memenuhi Kondisi Khusus

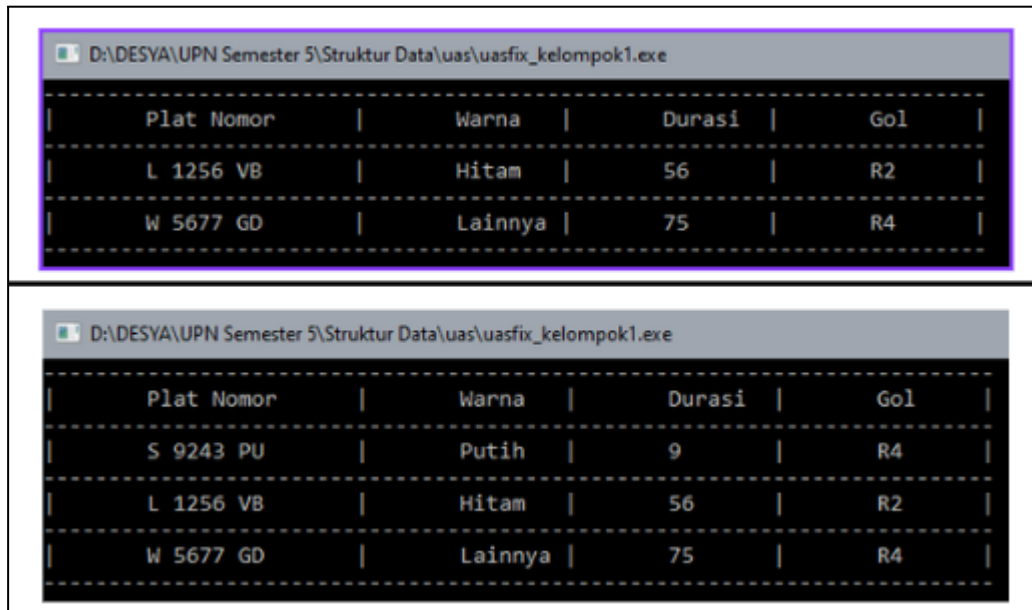
- Mendapat Diskon Ganjil Genap

```
C:\Users\ASUS\Documents\semester 3\strukdat\final project_sistemparkir.exe
No plat kendaraan: L 1213 UU
Biaya parkir kendaraan: 2500
Masukkan jumlah uang tunai: 5000
Kembalian: 2500

TERIMA KASIH TELAH MELAKUKAN PEMBAYARAN
Data dengan plat nomor kendaraan L 1213 UU telah terhapus!
```

Kendaraan di atas seharusnya membayar sebesar Rp 5000. Akan tetapi, dikarenakan tanggal penginputan data adalah ganjil dan plat nomor juga ganjil, maka pengendara tersebut mendapat diskon setengah harga menjadi Rp 2500.

- Mendapat Prioritas (Warna Kendaraan Sesuai dengan Ketentuan Hari Prioritas)



D:\DESYA\UPN Semester 5\Struktur Data\uas\uasfix_kelompok1.exe

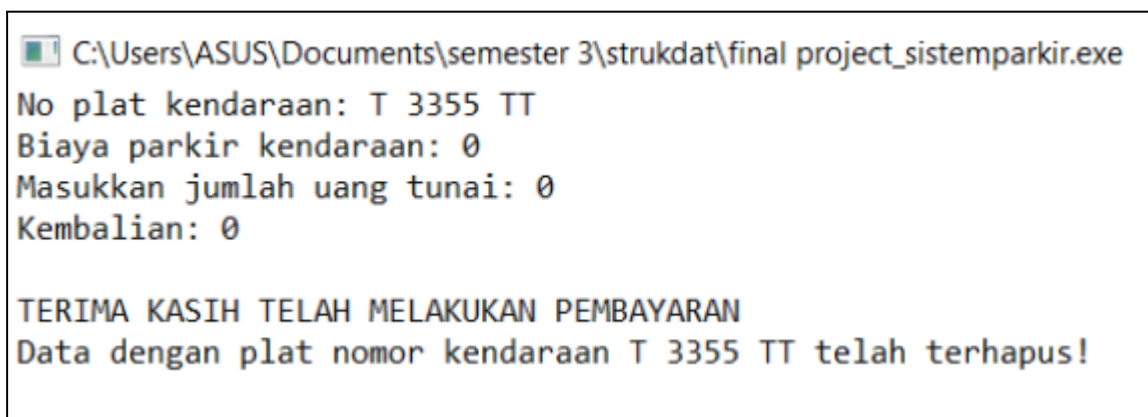
| Plat Nomor | Warna | Durasi | Gol |
|------------|---------|--------|-----|
| L 1256 VB | Hitam | 56 | R2 |
| W 5677 GD | Lainnya | 75 | R4 |

D:\DESYA\UPN Semester 5\Struktur Data\uas\uasfix_kelompok1.exe

| Plat Nomor | Warna | Durasi | Gol |
|------------|---------|--------|-----|
| S 9243 PU | Putih | 9 | R4 |
| L 1256 VB | Hitam | 56 | R2 |
| W 5677 GD | Lainnya | 75 | R4 |

Gambar pertama merupakan antrean sebelum data baru dimasukkan. Gambar kedua merupakan antrean setelah data baru dimasukkan. Hari penginputan data adalah hari Selasa yang mana kendaraan dengan warna putih akan mendapat prioritas menyela antrean sehingga data baru dengan plat nomor S 9243 PU di atas mendapat antrean pada baris pertama karena kendaraan juga berwarna putih.

- Durasi Parkir ≤ 5 Menit



C:\Users\ASUS\Documents\semester 3\strukdat\final project_sistemparkir.exe

No plat kendaraan: T 3355 TT
 Biaya parkir kendaraan: 0
 Masukkan jumlah uang tunai: 0
 Kembalian: 0

TERIMA KASIH TELAH MELAKUKAN PEMBAYARAN
 Data dengan plat nomor kendaraan T 3355 TT telah terhapus!

Kendaraan di atas parkir dengan durasi ≤ 5 menit sehingga pengendara tidak perlu membayar parkir (gratis).