

1 D4 - TEKKOM B

TUGAS PROYEK



Kelompok	: Kelompok 1
Anggota	: -Faisal Alief Hermawan (3221600045) -Septian Bagus Jumanoro (322160039) -Hamid Hamdani (3221600051) - Mohammad Haycal Elmo I.M (3221600042)
Kelas Dosen	: Ir. Sigit Wasista M.Kom
Mata Kuliah	: Praktikum Pemrograman Dasar 1
Hari/Tgl. Praktikum	: Jum'at, 5 November 2021



TUGAS PROYEK

SISTEM ANTRIAN PENGURUS SIM

A. TUJUAN

Sistem antrian Pelayanan SIM ini merupakan program yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi yang berfungsi untuk mempermudah customer dalam melakukan antrian pada pelayanan SIM di POLRES. Mekanisme dari sistem antrian Pelayanan SIM ini dimulai dari customer yang datang, kemudian memilih antrian untuk Buat SIM baru atau Perpanjangan SIM. Kemudian customer menginputkan Nama dan opsi jenis SIM yang dipilih, lalu data akan tersimpan. Kemudian customer menunggu dipanggil menurut antrian yang didapat. Setelah itu customer akan mendapat pelayanan, dan yang terakhir adalah customer melakukan pembayaran menurut nomer antrian yang didapat.

B. ALGORITMA

```
Program mulai
User memilih opsi
Opsi 1 :pilih opsi "buat sim baru"
           Pilih opsi "Perpanjangan SIM"
Masukkan Nama
Memilih opsi
Kembali ke menu y/n
Jika y kembali lagi ke awal
Jika n program selesai
Opsi 2
Memanggil antrian pertama
Kembali ke menu y/n
Jika y kembali lagi ke awal
Jika n program selesai

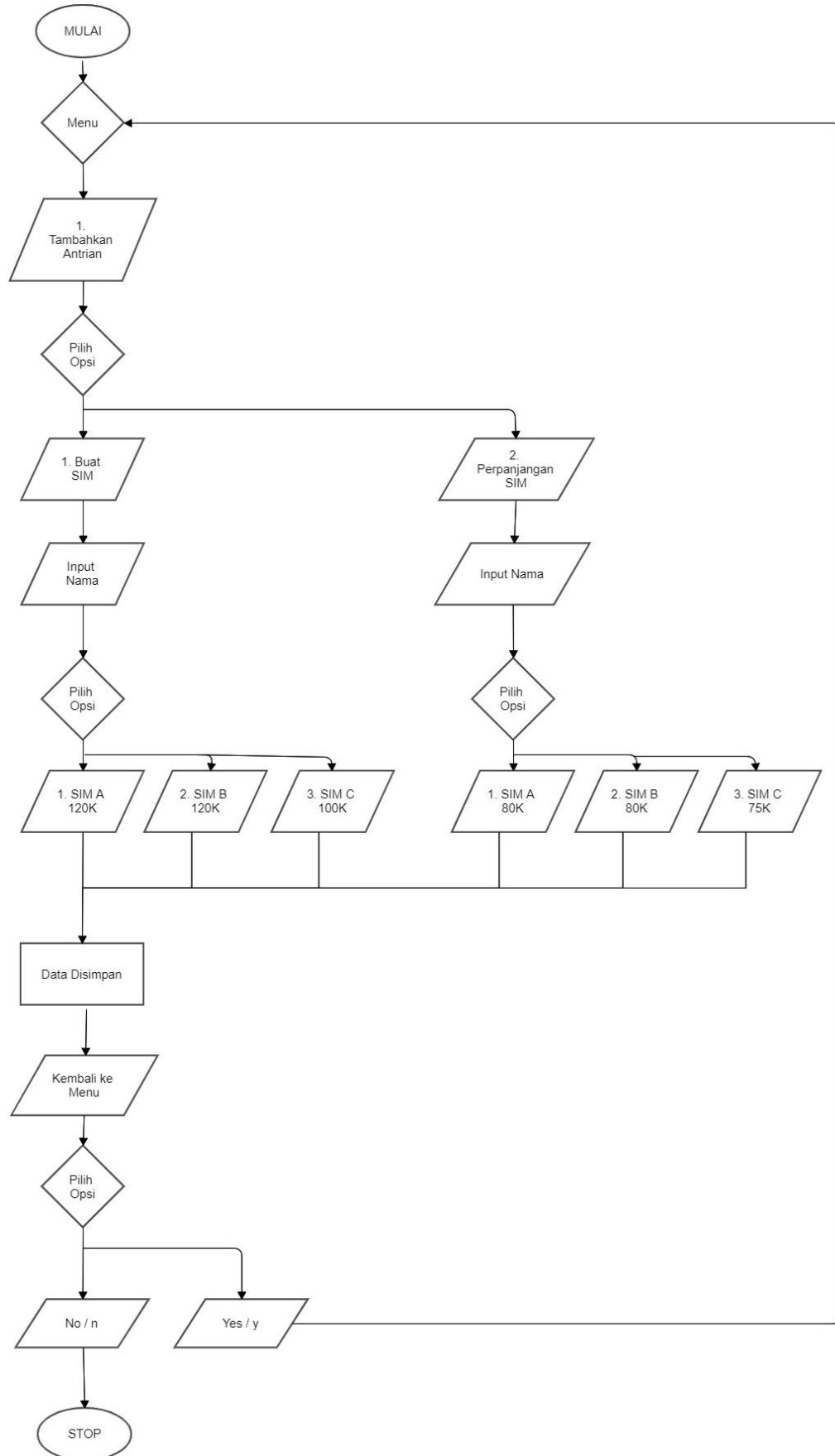
Opsi 3
Menampilkan daftar antrian+total bayar
Kembali ke menu y/n
Jika y kembali lagi ke awal
Jika n program selesai

Opsi 4
Total semua pendapatan
Kembali ke menu y/n
Jika y kembali lagi ke awal
Jika n program selesai

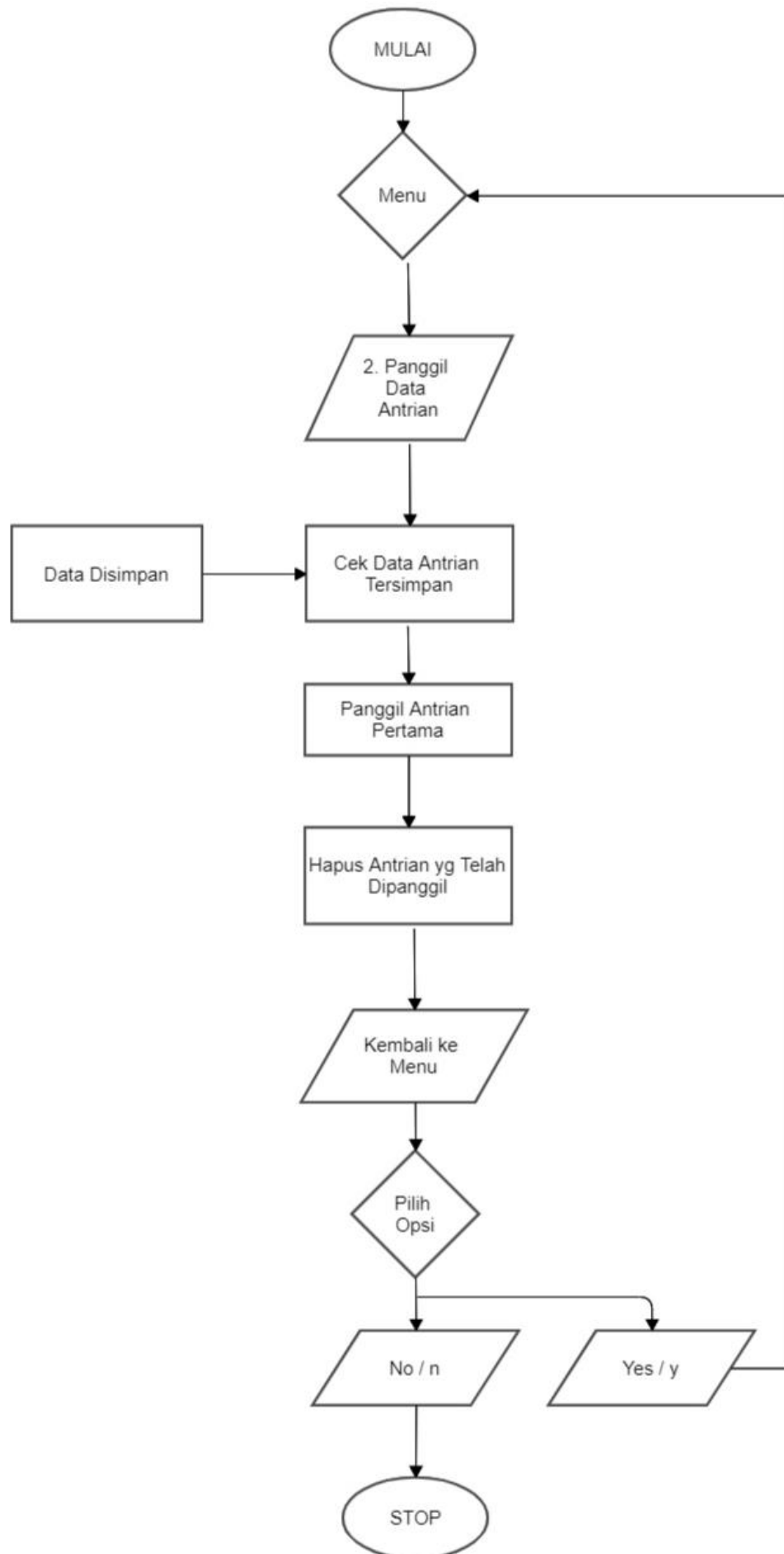
Opsi 5
Program selesai
```

C. FLOWCHART

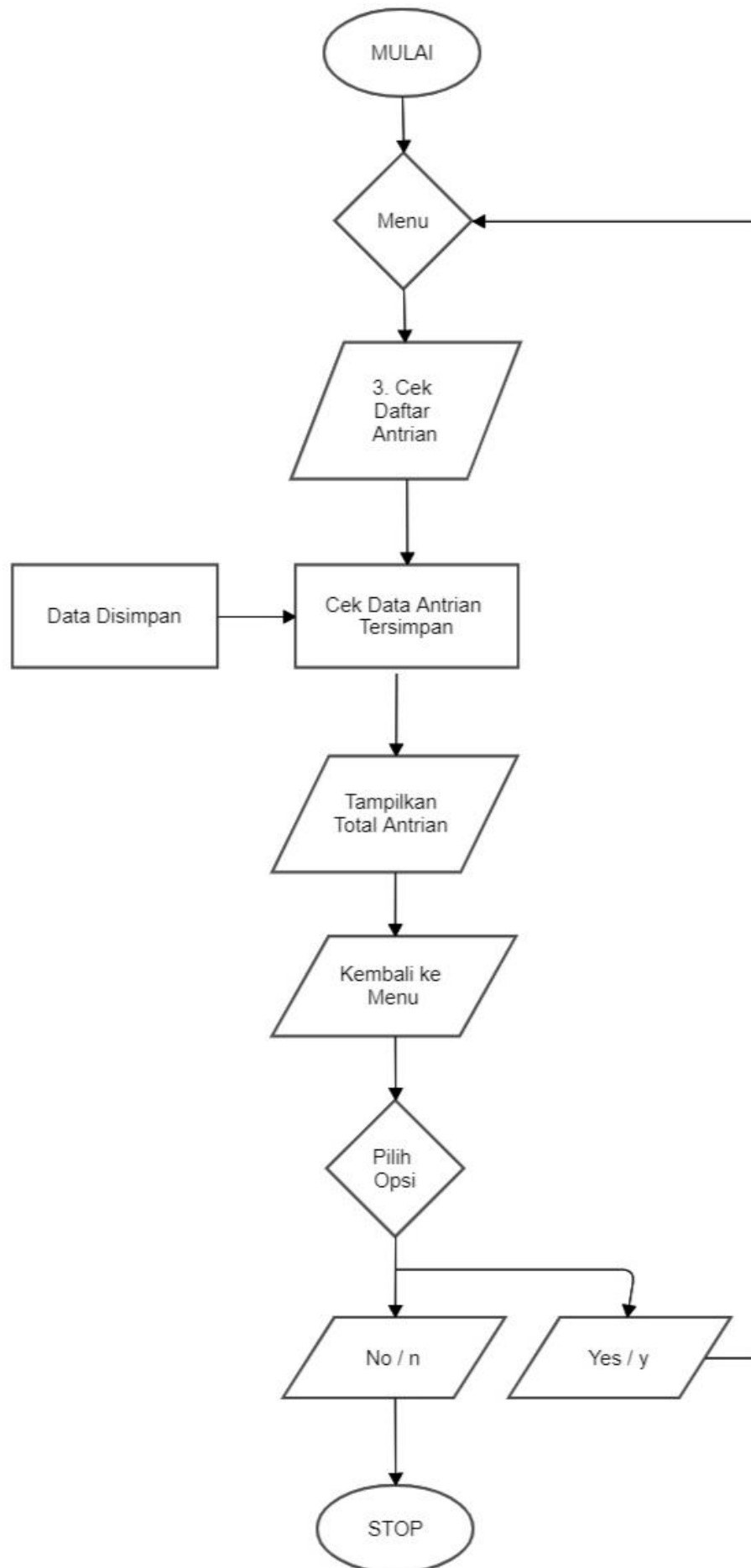
1. Menu 1



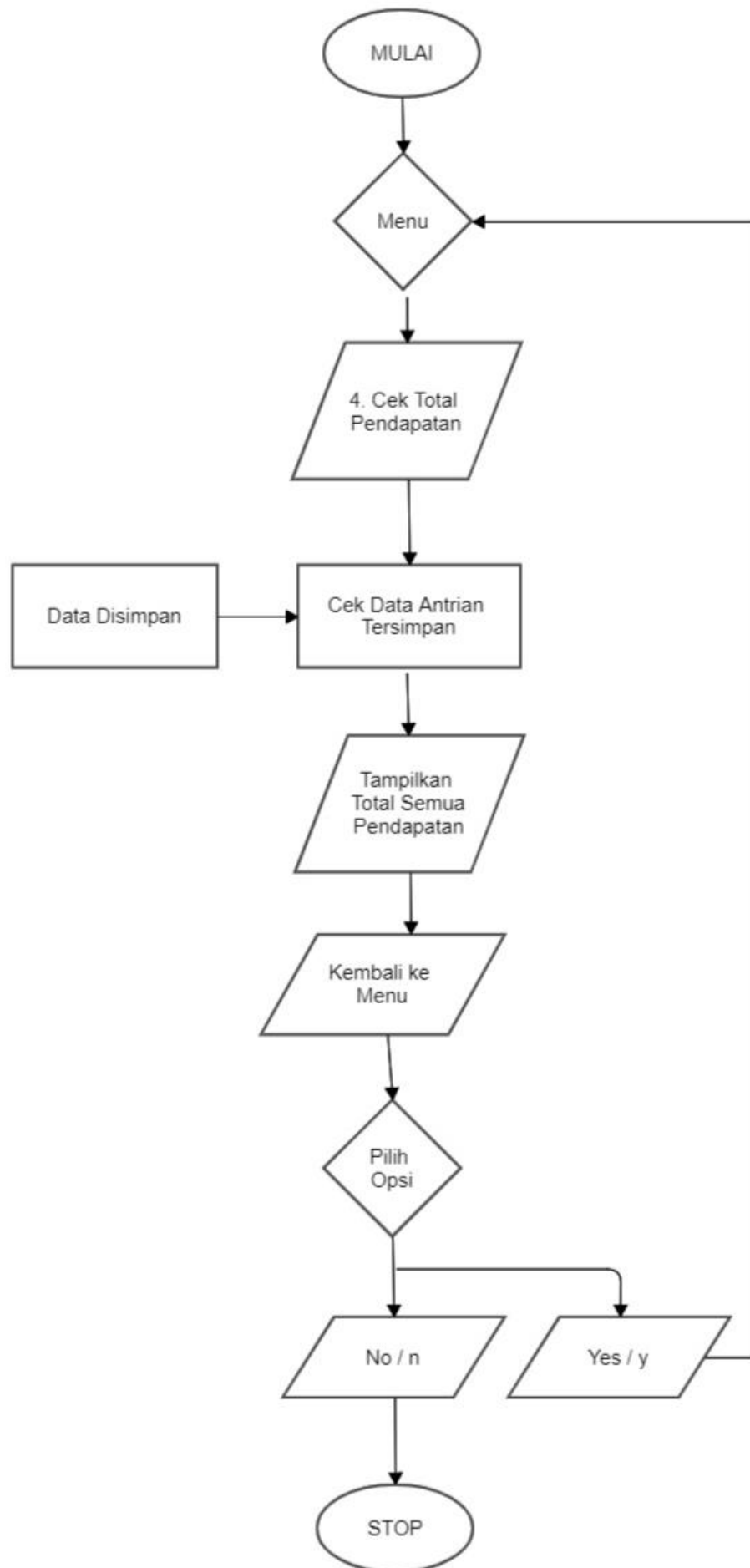
2. Menu 2



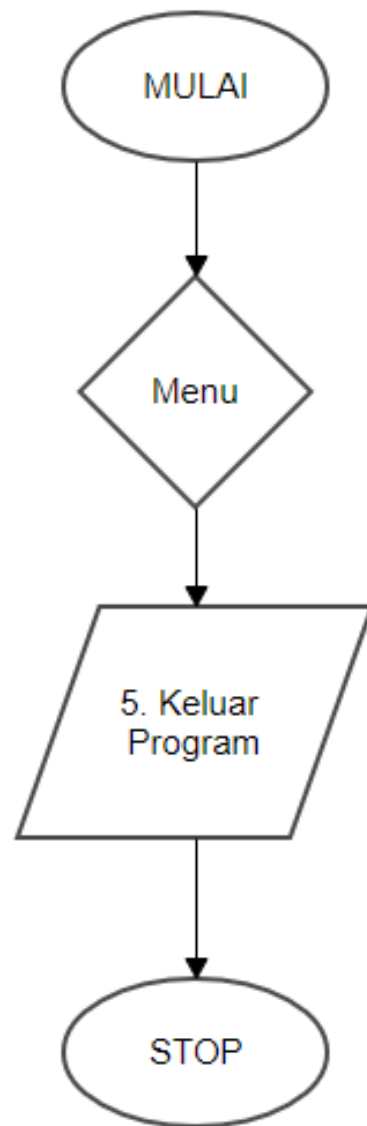
3. Menu 3



4. Menu 4



5. Menu 5



D. SOURCE CODE

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
#include<stdlib.h>
#define max 50
#define panjang 100

struct data{
    char nama[max][panjang];
    char cek[max][panjang];
    int cari[max]={0};
    long long int total=0;
    const long int sima[2]={120000,80000};
    const long int simb[2]={120000,80000};
    const long int simc[2]={100000,75000};
    const char tipe[3][max]={"SIM A", "SIM B", "SIM C"};
    int head=0;
    int tail=0;
    int menu[max]={0};
}d;
int pendapatan(int menu,int pilih){
    if(menu==1){
        if(pilih==1){d.total+=d.sima[menu-1];}
        else if(pilih==2){d.total+=d.simb[menu-1];}
        else if(pilih==3){d.total+=d.simc[menu-1];}
        puts("\n[Data Telah Dimasukkan Dalam Antrian]\n");
    }
}
```



```

        else if(menu==2){
            if(pilih==1){d.total+=d.sima[menu-1];}
            else if(pilih==2){d.total+=d.simb[menu-1];}
            else if(pilih==3){d.total+=d.simc[menu-1];}
            puts("\n[Data Telah Dimasukkan Dalam Antrian]\n");
        }
    }

int antrian(){
    int pilih,menu;

    puts("=====");
    puts("1.  Buat SIM Baru");
    puts("2.  Perpanjangan SIM");
    puts("=====");
    printf("Silahkan Pilih Keperluan Anda: ");
    scanf("%d", &menu);
    d.menu[d.tail]=menu;
    puts("Menambahkan Data Antrian");
    printf("Atas Nama: ");
    //printf("%d",d.tail);
    getchar();scanf("%s", &d.nama[d.tail]);
    puts("=====");
    printf("1. SIM A      ||      Rp. %d\n",d.sima[menu-1]);
    printf("2. SIM B      ||      Rp. %d\n",d.simb[menu-1]);
    printf("3. SIM C      ||      Rp. %d\n",d.simc[menu-1]);
    puts("=====");
    printf("Keperluan Mengurus: ");
    scanf("%d", &pilih);
    d.cari[d.tail]=pilih;
    //printf("\n %d %d %d %d %d",d.cari[d.tail],d.tail,d.menu,pilih);
    d.tail++;
    pendapatan(menu,pilih);
}

int panggil(){
    if(d.tail==0){puts("Antrian Kosong");}
    else{
        puts("");
        puts("Antrian Yang Dipanggil Atas Nama: ");
        puts("");
        printf("[%s dengan keperluan mengurus %s]\n\n", d.nama[d.head],d.tipe[d.cari[d.head]-1]);
        printf("[Silahkan Menuju Loker Pembayaran]\n");
        for(int i=d.head;i<d.tail;i++){
            strcpy(d.nama[i],d.nama[i+1]);
            d.cari[i]=d.cari[i+1];
            d.menu[i]=d.menu[i+1];
        }
        d.tail--;
    }
}

}

```

```

int check(){
    int total[max]={0};
    if(d.tail==0){puts("Antrian Kosong");}
    else{
        for(int i=0;i<d.tail;i++){
            //printf("%d %d\n",d.cari[i],d.tail);
            if(d.cari[i]==1){
                total[i]+=d.sima[d.menu[i]-1];
                printf("Nama: %s\n",d.nama[i]);
                printf("Harga yang harus dibayar: Rp.%d\n",total[i]);
            }
            else if(d.cari[i]==2){
                total[i]+=d.simb[d.menu[i]-1];
                printf("Nama: %s\n",d.nama[i]);
                printf("Harga yang harus dibayar: Rp.%d\n",total[i]);
            }
            else if(d.cari[i]==3){
                total[i]+=d.simc[d.menu[i]-1];
                printf("Nama: %s\n",d.nama[i]);
                printf("Harga yang harus dibayar: Rp.%d\n",total[i]);
            }
        }
    }
}

int main(){
    int pilih,pilihan;
    do{
        system("cls");
        puts("=====");
        puts("    Antrian Mengurus SIM Lendaraan    ");
        puts("=====");
        puts("1.  Tambahkan Antrian");
        puts("2.  Panggil Data Antrian");
        puts("3.  Cek Daftar Antrian");
        puts("4.  Cetak Total Pendapatan");
        puts("5.  Keluar Dari Program");
        puts("=====");
        printf("Masukkan Pilihan Anda: ");
        scanf("%d", &pilih);

        switch(pilih){
            case 1:{
                antrian();
                break;
            }
            case 2:{
                panggil();
                break;
            }
            case 3:{
                check();
                break;
            }
        }
    }
}

```

```

case 4:{
    printf("\nTotal Pendapatan kali ini: Rp.%d",d.total);
    puts("");
    break;
}
case 5:{
    return 0;
    break;
}
}
puts("");
printf("Kembali ke menu y/n: ");
pilihan=getche();

}while(pilihan!='n');
}

```

E. OUTPUT

- Menu 1

```

=====
                Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 1
=====
1.  Buat SIM Baru
2.  Perpanjangan SIM
=====
Silahkan Pilih Keperluan Anda: 1
Menambahkan Data Antrian
Atas Nama: Septian
=====
1. SIM A      ||      Rp. 120000
2. SIM B      ||      Rp. 120000
3. SIM C      ||      Rp. 100000
=====
Keperluan Mengurus: 1

[Data Telah Dimasukkan Dalam Antrian]

Kembali ke menu y/n: █

```

- Menu 2

```
=====
          Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 2

Antrian Yang Dipanggil Atas Nama:

[Septian dengan keperluan mengurus SIM A]

[Silahkan Menuju Loker Pembayaran]

Kembali ke menu y/n: █
```

- Menu 3 dengan kondisi data antrian kosong

```
=====
          Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 3
Antrian Kosong

Kembali ke menu y/n: █
```

- Menu 3 dengan kondisi terdapat data antrian

```
=====
          Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 3
Nama: Septian
Harga yang harus dibayar: Rp.120000
Nama: Faisal
Harga yang harus dibayar: Rp.80000
Nama: Hamid
Harga yang harus dibayar: Rp.100000
Nama: Haycal
Harga yang harus dibayar: Rp.75000

Kembali ke menu y/n: █
```

- Menu 4 dengan kondisi data antrian Menu 3 terdapat data antrian

```
=====
          Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 4

Total Pendapatan kali ini: Rp.375000

Kembali ke menu y/n: █
```

F. ANALISA

Pada project kali ini kami membuat sebuah program Antrian Mengurus SIM Kendaraan, dimana pada program tersebut terdapat beberapa menu untuk pengguna yang ingin mengurus SIM baru atau memperpanjang masa aktif dari SIM. Pada menu yang dipilih juga telah terdapat harga sesuai dengan harga yang ada pada SAMSAT.

Untuk source code sebagai berikut:

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

```

#define max 50
#define panjang 100

struct data{
    char nama[max][panjang];
    char cek[max][panjang];
    int cari[max]={0};
    long long int total=0;
    const long int sima[2]={120000,80000};
    const long int simb[2]={120000,80000};
    const long int simc[2]={100000,75000};
    const char tipe[3][max]={"SIM A", "SIM B", "SIM C"};
    int head=0;
    int tail=0;
    int menu[max]={0};
}d;

```

Pada header program menggunakan struct data agar lebih terstruktur yang disimpan pada variabel d. Disini telah didefinisikan bahwa variabel max 50 dan panjang 100. Lalu untuk harga dari pengurusan sim tersebut disimpan dalam variabel array yaitu sima, simb, dan simc.

```

int antrian(){
    int pilih,menu;

    puts("=====");
    puts("1.    Buat SIM Baru");
    puts("2.    Perpanjangan SIM");
    puts("=====");
    printf("Silahkan Pilih Keperluan Anda: ");
    scanf("%d", &menu);
    d.menu[d.tail]=menu;
    puts("Menambahkan Data Antrian");
    printf("Atas Nama: ");
    //printf("%d",d.tail);
    getchar();scanf("%s", &d.nama[d.tail]);
    puts("=====");
    printf("1. SIM A      ||      Rp. %d\n",d.sima[menu-1]);
    printf("2. SIM B      ||      Rp. %d\n",d.simb[menu-1]);
    printf("3. SIM C      ||      Rp. %d\n",d.simc[menu-1]);
    puts("=====");
    printf("Keperluan Mengurus: ");
    scanf("%d", &pilih);
    d.cari[d.tail]=pilih;
    //printf("\n %d %d %d %d",d.cari[d.tail],d.tail,d.menu,pilih);
    d.tail++;
    pendapatan(menu,pilih);
}

```

Selanjutnya masuk pada fungsi antrian, dimana user akan memasukkan pilihan beserta namanya dan disimpan dalam data. Lalu user memilih sesuai nomor dan disimpan dalam pilih. Setelah itu disimpan dalam array d.tail dimana d.tail tersebut akan bertambah terus atau d.tail++. Untk tampilan outputnya seperti berikut:

```

=====
          Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 1
=====
1.  Buat SIM Baru
2.  Perpanjangan SIM
=====
Silahkan Pilih Keperluan Anda: 1
Menambahkan Data Antrian
Atas Nama: Septian
=====
1. SIM A      ||      Rp. 120000
2. SIM B      ||      Rp. 120000
3. SIM C      ||      Rp. 100000
=====
Keperluan Mengurus: 1

[Data Telah Dimasukkan Dalam Antrian]

Kembali ke menu y/n: █

```

```

int panggil(){
if(d.tail==0){puts("Antrian Kosong");}
else{
    puts("");
    puts("Antrian Yang Dipanggil Atas Nama: ");
    puts("");
    printf("[%s dengan keperluan mengurus %s]\n\n",
d.nama[d.head],d.tipe[d.cari[d.head]-1]);
    printf("[Silahkan Menuju Loker Pembayaran]\n");
    for(int i=d.head;i<d.tail;i++){
        strcpy(d.nama[i],d.nama[i+1]);
        d.cari[i]=d.cari[i+1];
        d.menu[i]=d.menu[i+1];
    }
    d.tail--;
}
}

```


Selanjutnya fungsi panggil, dimana algoritmanya menganut algoritma FIFO atau First In First Out yang berarti data yang diinputkan lalu disimpan pertama kali maka data tersebut yang akan keluar pertama juga. Terdapat statement jika `d.tail == 0`, atau tidak ada data yang diinputkan maka akan mencetak "Antrian Kosong". Jika terdapat data yang diinputkan maka akan tercetak nama beserta keperluan yang telah diinputkan.

Untuk outputnya sebagai berikut:

```
=====
      Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 2

Antrian Yang Dipanggil Atas Nama:

[Septian dengan keperluan mengurus SIM A]

[Silahkan Menuju Loker Pembayaran]

Kembali ke menu y/n: █
```

```
=====
      Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 3
Antrian Kosong

Kembali ke menu y/n: █
```

```
int check(){
    int total[max]={0};
    if(d.tail==0){puts("Antrian Kosong");}
    else{
        for(int i=0;i<d.tail;i++){
            //printf("%d %d\n",d.cari[i],d.tail);
            if(d.cari[i]==1){
                total[i]+=d.sima[d.menu[i]-1];
                printf("Nama: %s\n",d.nama[i]);
                printf("Harga yang harus dibayar: Rp.%d\n",total[i]);
            }
            else if(d.cari[i]==2){
                total[i]+=d.simb[d.menu[i]-1];
                printf("Nama: %s\n",d.nama[i]);
                printf("Harga yang harus dibayar: Rp.%d\n",total[i]);
            }
            else if(d.cari[i]==3){
                total[i]+=d.simc[d.menu[i]-1];
                printf("Nama: %s\n",d.nama[i]);
                printf("Harga yang harus dibayar: Rp.%d\n",total[i]);
            }
        }
    }
}
```



```
}
}
```

Selanjutnya fungsi check, yang berfungsi untuk mengecek data yang disimpan. Sama seperti fungsi panggil, dimana jika `d.tail == 0` maka akan tercetak "Antrian Kosong". Jika terdapat data yang tersimpan maka, program akan melanjutkan ke dalam if bercabang. Dimana data akan dicek satu persatu apabila memenuhi maka akan disimpan sesuai dengan pilihan keperluan dan juga akan tampil harganya. Untuk harga dipanggil dari variabel total yang disimpan pada array `[i]`. Untuk tampilan outputnya sebagai berikut:

```
=====
          Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 3
Nama: Septian
Harga yang harus dibayar: Rp.120000
Nama: Faisal
Harga yang harus dibayar: Rp.80000
Nama: Hamid
Harga yang harus dibayar: Rp.100000
Nama: Haycal
Harga yang harus dibayar: Rp.75000
Kembali ke menu y/n: █
```

```
int pendapatan(int menu,int pilih){
    if(menu==1){
        if(pilih==1){d.total+=d.sima[menu-1];}
        else if(pilih==2){d.total+=d.simb[menu-1];}
        else if(pilih==3){d.total+=d.simc[menu-1];}
        puts("\n[Data Telah Dimasukkan Dalam Antrian]\n");
    }
    else if(menu==2){
        if(pilih==1){d.total+=d.sima[menu-1];}
        else if(pilih==2){d.total+=d.simb[menu-1];}
        else if(pilih==3){d.total+=d.simc[menu-1];}
        puts("\n[Data Telah Dimasukkan Dalam Antrian]\n");
    }
}
```

Selanjutnya fungsi pendapatan, dimana total pendapatan disimpan dalam variabel `d.total`. Terdapat beberapa statement yang merujuk pada `d.total` yang selanjutnya akan ditambahkna sesuai dengan menu yang telah dipilih dan disimpan dalam array `[i]`.

Untuk tampilan outputnya sebagai berikut:

```
=====
      Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda: 4

Total Pendapatan kali ini: Rp.375000

Kembali ke menu y/n: █
```

```
int main(){
    int pilih,pilihan;
    do{
        system("cls");
        puts("=====");
        puts("      Antrian Mengurus SIM Kendaraan      ");
        puts("=====");
        puts("1.  Tambahkan Antrian");
        puts("2.  Panggil Data Antrian");
        puts("3.  Cek Daftar Antrian");
        puts("4.  Cetak Total Pendapatan");
        puts("5.  Keluar Dari Program");
        puts("=====");
        printf("Masukkan Pilihan Anda: ");
        scanf("%d", &pilih);

        switch(pilih){
            case 1:{
                antrian();
                break;
            }
            case 2:{
                panggil();
                break;
            }
            case 3:{
                check();
                break;
            }
            case 4:{
                printf("\nTotal Pendapatan kali ini: Rp.%d",d.total);
                puts("");
                break;
            }
        }
    }
```

```

        case 5:{
            return 0;
            break;
        }
    }
    puts("");
    printf("Kembali ke menu y/n: ");
    pilihan=getche();

}while(pilihan!='n');
}

```

Selanjutnya fungsi main, didalam fungsi tersebut terdapat perulangan do-while sehingga program akan terus berjalan hingga pilihan != 'n'. Lalu juga terdapat switch case yang merupakan syntax untuk menentukan pilihan, dengan menginputkan nomor 1-5. Jika user menginputkan pilihan ke 5, maka program akan berhenti karena return 0. Untuk tampilan outputnya sebagai berikut:

```

=====
          Antrian Mengurus SIM Kendaraan
=====
1.  Tambahkan Antrian
2.  Panggil Data Antrian
3.  Cek Daftar Antrian
4.  Cetak Total Pendapatan
5.  Keluar Dari Program
=====
Masukkan Pilihan Anda:

```

G. KESIMPULAN

Dari program antrian sim kendaraan ini dapat disimpulkan bahwa antrian dengan menggunakan program seperti ini jauh lebih efektif dibandingkan dengan antrian manual yang menggunakan nomor antrian dan mengisi data diri di sebuah form kertas yang di berikan. Dengan program antrian ini, petugas cukup memasukkan data diri orang tersebut. Selanjutnya orang tersebut akan terdaftar dalam antrian secara otomatis dan datanya pun juga aman disimpan dalam program.