Tugas Pendahuluan VI Makassar, 25 November 2024

**PRAKTIKUM**

**STRUKTUR DATA**



Nama : Andi Ikhlas Mallomo

Stambuk : 13020230251

Frekuensi : TI\_SD-7 (A7)

Dosen : Syariful Mujaddid, S.Kom., M.T.

Asisten 1 : Annisa Pratama Putri

Asisten 2 : Rahma Puspitasari

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS MUSLIM INDONESIA**

**MAKASSAR**

**2024**

1.Buatlah kodingan fitur project yang menjadi tanggung jawabmu

* **A computer chip with blue lines and dots

  Description automatically generatedFITUR MEMASUKKAN MOBIL**

#include <iostream>

using namespace std;

const int MAX\_SLOT = 1;

const int MAX\_QUEUE = 1;

string slotParkir[MAX\_SLOT];

string waktuParkir[MAX\_SLOT];

int jumlahSlotTerisi = 0;

string waktuAntrean[MAX\_SLOT];

string antreanMobil[MAX\_QUEUE];

int frontAntrean = 0, rearAntrean = -1, jumlahAntrean = 0;

int pendapatan = 0;

string getWaktuSekarang() {

time\_t now = time(0);

tm \*ltm = localtime(&now);

char buffer[20];

strftime(buffer, sizeof(buffer), "%Y-%m-%d %H:%M:%S", ltm);

return string(buffer);

}

**A computer chip with blue lines and dots

Description automatically generated**void garisHorizontal(int panjang) {

for (int i = 0; i < panjang; i++) {

cout << "-";

}

cout << endl;

}

void masukkanMobil() {

if (jumlahSlotTerisi < MAX\_SLOT) {

string platNomor;

garisHorizontal(41);

cout << "| MENAMBAHKAN MOBIL |\n";

garisHorizontal(41);

cout << "Masukkan plat nomor mobil: ";

getline(cin, platNomor);

slotParkir[jumlahSlotTerisi] = platNomor;

waktuParkir[jumlahSlotTerisi] = getWaktuSekarang();

jumlahSlotTerisi++;

cout << "\n\n";

cout << "++++++++++++++++++++++++++++++++++++++++++\n";

cout << "+ Mobil Plat : " << platNomor << "\n";

cout << "+ Date : " << getWaktuSekarang() << "\n";

cout << "++++++++++++++++++++++++++++++++++++++++++\n\n";

**A computer chip with blue lines and dots

Description automatically generated**

cout << "berhasil diparkir.\n";

} else {

if (jumlahAntrean < MAX\_QUEUE) {

string platNomor;

cout << "Parkir penuh. Masukkan plat nomor mobil ke antrean: ";

getline(cin, platNomor);

rearAntrean++;

antreanMobil[rearAntrean] = platNomor;

waktuAntrean[rearAntrean] = getWaktuSekarang();

jumlahAntrean++;

cout << "\n\n";

cout << "++++++++++++++++++++++++++++++++++++++++++\n";

cout << "+ Mobil Plat : " << platNomor << "\n";

cout << "+ Date : " << getWaktuSekarang() << "\n";

cout << "++++++++++++++++++++++++++++++++++++++++++\n\n";

} else {

cout << "Parkir penuh dan antrean juga penuh. Tidak dapat menerima mobil lagi.\n";

}

}

}

int main() {

**A computer chip with blue lines and dots

Description automatically generated** masukkanMobil();

}

A screen shot of a computer

Description automatically generated

* **FITUR MENGELUARKAN MOBIL**

#include <iostream>

using namespace std;

const int MAX\_SLOT = 1; // Kapasitas maksimum slot parkir

const int MAX\_QUEUE = 1; // Kapasitas maksimum antrean

const int MAX\_RIWAYAT = MAX\_SLOT \* 2; // Kapasitas maksimum riwayat

string slotParkir[MAX\_SLOT];

string waktuParkir[MAX\_SLOT];

**A computer chip with blue lines and dots

Description automatically generated**int jumlahSlotTerisi = 0;

string riwayatParkir[MAX\_RIWAYAT];

string waktuMasukRiwayat[MAX\_RIWAYAT];

string waktuKeluarRiwayat[MAX\_RIWAYAT];

int jumlahRiwayat = 0;

string waktuAntrean[MAX\_SLOT];

string antreanMobil[MAX\_QUEUE];

int frontAntrean = 0, rearAntrean = -1, jumlahAntrean = 0;

int pendapatan = 0;

string getWaktuSekarang() {

time\_t now = time(0);

tm \*ltm = localtime(&now);

char buffer[20];

strftime(buffer, sizeof(buffer), "%Y-%m-%d %H:%M:%S", ltm);

return string(buffer);

}

void garisHorizontal(int panjang) {

for (int i = 0; i < panjang; i++) {

cout << "-";

}

**A computer chip with blue lines and dots

Description automatically generated** cout << endl;

}

void keluarkanMobil() {

garisHorizontal(41);

cout << "| MENGELUARKAN MOBIL |\n";

garisHorizontal(41);

if (jumlahSlotTerisi > 0) {

string platNomor;

cout << "Masukkan plat nomor mobil yang akan dikeluarkan: ";

getline(cin, platNomor);

bool ditemukan = false;

for (int i = 0; i < jumlahSlotTerisi; i++) {

if (slotParkir[i] == platNomor) {

ditemukan = true;

// Tambahkan ke riwayat

riwayatParkir[jumlahRiwayat] = slotParkir[i];

waktuMasukRiwayat[jumlahRiwayat] = waktuParkir[i];

waktuKeluarRiwayat[jumlahRiwayat] = getWaktuSekarang();

jumlahRiwayat++;

// Geser slot parkir

for (int j = i; j < jumlahSlotTerisi - 1; j++) {

**A computer chip with blue lines and dots

Description automatically generated** slotParkir[j] = slotParkir[j + 1];

waktuParkir[j] = waktuParkir[j + 1];

}

jumlahSlotTerisi--;

pendapatan += 5000;

cout << "\n\n";

cout << "++++++++++++++++++++++++++++++++++++++++++\n";

cout << "+ Mobil Plat : " << platNomor << "\n";

cout << "+ Date : " << getWaktuSekarang() << "\n";

cout << "++++++++++++++++++++++++++++++++++++++++++\n\n";

// Jika ada antrean, tambahkan ke slot parkir

if (jumlahAntrean > 0) {

slotParkir[jumlahSlotTerisi] = antreanMobil[frontAntrean];

waktuParkir[jumlahSlotTerisi] = getWaktuSekarang();

jumlahSlotTerisi++;

for (int i = frontAntrean; i < rearAntrean; i++) {

antreanMobil[i] = antreanMobil[i + 1];

waktuAntrean[i] = waktuAntrean[i + 1];

}

frontAntrean++;

jumlahAntrean--;

}

break;

}

}

if (!ditemukan) {

cout << "[" << getWaktuSekarang() << "] Mobil dengan plat " << platNomor << " tidak ditemukan di slot parkir.\n";

}

} else {

cout << "[" << getWaktuSekarang() << "] Tidak ada mobil di slot parkir.\n";

}

}

int main() {

keluarkanMobil();

}

A black and white sign with white text

Description automatically generated

**A computer chip with blue lines and dots

Description automatically generated**

**A computer chip with blue lines and dots

Description automatically generated**