Program ini berisi dua loop dan satu fungsi rekursif.

Loop adalah struktur aliran kontrol yang memungkinkan pemrogram mengulang blok kode beberapa kali atau hingga kondisi tertentu terpenuhi.

Dalam program ini, loop pertama adalah loop for yang mengulang setiap kamar yang telah dipesan pengguna.

Untuk setiap ruangan, program meminta pengguna untuk memasukkan nomor kamar, tipe, jumlah tamu, dan tarif.

Loop kedua juga merupakan loop for, tetapi bersarang di dalam loop pertama. Loop ini digunakan untuk meminta pengguna memasukkan nama tamu yang menginap di setiap kamar.

Jumlah iterasi loop ini ditentukan oleh nilai numberOfGuests untuk setiap kamar.

Fungsi rekursif adalah fungsi yang memanggil dirinya sendiri. Dalam program ini, fungsi countTotalCost() adalah fungsi rekursif. Dibutuhkan empat argumen: jumlah total kamar, jumlah malam, tarif kamar, dan kamar saat ini.

Fungsi menghitung total biaya pemesanan dengan memanggil dirinya sendiri dengan kamar berikutnya hingga semua kamar telah diproses, dan mengembalikan 0.

Hasil akhirnya adalah jumlah biaya setiap kamar.

bagian loop dan nested loop:

for (int i = 1; i <= numberOfRooms; i++)

{

cout << "\tSilakan masukkan no kamar " << i << ":";

cin >> roomNumber;

cout << "\tMau Pilih Tipe Apa (single/double/suite): ";

cin >> roomType;

cout << "\tHarap masukkan jumlah tamu untuk kamar " << i << ": ";

cin >> numberOfGuests;

cout << "\tharga Tersedia saat ini (10/20/50) Silakan Pilih " << "Untuk Kamar "<< i << ": ";

cin >> roomRate;

// menngulang setiap ruangan

for (int j = 1; j <= numberOfGuests; j++)

{

cout << "\n\tSilakan masukkan nama tamu: \n";

string guestName;

cout << "\tTamu " << j << ": " << guestName;

Bagian Rekrusifnya:

Fungsi countTotalCost() bersifat rekursif. Itu memanggil dirinya sendiri dengan argumen yang diperbarui (currentRooms + 1)

setiap kali dipanggil, hingga currentRooms lebih besar dari numberOfRooms, dalam hal ini mengembalikan 0.

Ini memungkinkan fungsi untuk menghitung biaya total semua kamar dalam pemesanan dengan menambahkan biaya setiap kamar pada gilirannya.

Berikut adalah kode untuk fungsi countTotalCost() :

double calculateTotalCost(int numberOfRooms, int numberOfNights, double roomRate, int currentRooms)

{

// Jika kita telah mengulang semua kamar, kembalikan 0

if (currentRooms > numberOfRooms)

{

return 0;

}

else {

// Jika tidak, tambahkan biaya kamar saat ini ke total dan panggil fungsi lagi dengan kamar berikutnya

return calculateTotalCost(numberOfRooms, numberOfNights, roomRate, currentRooms + 1) + numberOfNights \* roomRate;

}

}

Dalam fungsi main(), fungsi ini disebut sebagai berikut:

totalCost = calculateTotalCost(numberOfRooms, numberOfNights, roomRate, 1);

Angka 1 dalam panggilan ke countTotalCost() menunjukkan bahwa penghitungan harus dimulai dengan ruangan pertama.

Fungsi kemudian akan memanggil dirinya sendiri dengan currentRooms diatur ke 2, lalu 3, dan seterusnya,

sampai telah menambahkan biaya semua kamar.