

# Hotelli Varauspalvelu

Hotellin varauspalvelu toimii seuraavalla periaatteella.

Aluksi määritetään mitä kaikkia eri datatyppejä tarvitaan varauspalvelun suorittamiseen.

```
#include <iostream>
#include <cstdlib>
#include <ctime>
#include <vector>
#include <algorithm>
#include <string>
```

Iostreamilla saadaan cout ja cin eli ulossyöttö ja sisäänsyöttö.

Cstdlib avulla saadaan rand ja srand eli satunnaisten luvun tekijän ja satunnaisien luvun tietojen käsittelyä.

Ctime tuo tietokoneen ajan.

Vector:illa saadaan luotua vektoreita ja luettua niitä.

Algorithm avulla saadaan etsittyä ettei tule samoja vektoreiden sisään.

String avulla saadaan henkilön nimi talteen.

Kun lisätään myös koodi: using namespace std, niin saadaan koodista selkeää ja ei tarvitse laittaa std::

```
static int hinta() {
    const int maxH = 100;
    const int minH = 80;
    return minH + rand() % (maxH - minH + 1);
}
```

Tässä kohdassa luodaan aliohjelma nimeltä: hinta. Tällä aliohjelmalla saadaan selvitettyä yhden huoneen hinta per yö. Määritettiin maksimihinta nimellä maxH ja minimihinta minH. Lasketaan hinta ja joka kerta kun suoritetaan ohjelmaa tulee uusi arvottu hinta.

```
int main() {
    //Aluksi muodostetaan kaikki muuttujat joita tarvitaan
    int haluttuja_huoneita;
    vector<int> halutut_huoneet;
    const int max = 70;
    const int min = 30;
    string nimi;
    char valinta = 0;
    int yot = 0;
```

Aloitetaan itse pääohjelma main() ja määritetään kaikki muuttajat joita tullaan tarvitsemaan tässä ohjelmassa.

```
//Jotta voidaan aloittaa ohjelma uudelleen on oltava do koodin alussa
do {
    //Tyhjennetään konsoli ruutu jos aloitetaan alusta
    system("cls");

    srand(time(0));
```

Jotta voidaan lopussa aloittaa koko ohjelma uudestaan tarvitaan do. System("cls") tyhjentää konsoliruudun jotta voidaan aloittaa alusta konsoliruutu. Srand(time(0)) avulla saadaan määritettyä kuinka kauan satunnaisten luvun muodostamiseen menee.

```
//Arvotan huoneiden maara valilla 30-70
int kaikki_huoneet = min + rand() % (max - min + 1);
```

Arvotaan kaikki huoneet väliltä 30-70 joka määrättiin tehtävässä.

```
//Muodostetaan varatut huoneet
vector<int> varatut_huoneet;

//Arvotaan kuinka monta huonetta on varattu valilla 1 - mitä ikäna int kaikki antaa
const int varatut = rand() % 21 + 1;
```

Muodostetaan vektori varatuille huoneille jotta voidaan tarkastella sitä myöhemmin. Arvotaan tänne varatuille vektorille huoneet jotka ovat varattuja. Varattuja huoneita on 1-20 välillä.

```
//Arvotaan huoneiden numerot
for (int i = 0; i < varatut; i++) {
    int random_varatut = rand() % kaikki_huoneet + 1;

    //Tarkistetaan ettei ole tupla varauksia
    if (find(varatut_huoneet.begin(), varatut_huoneet.end(), random_varatut) == varatut_huoneet.end()) {
        varatut_huoneet.push_back(random_varatut);
    }
}
```

Arvotaan näiden huoneiden numerot jotka ovat varattuja. Jottei tulisi vahingossa samaa huonetta varattua useampaan kertaan tarkasteellan se vektorin sisältä find koodin avulla. Se käy läpi vektorin ja tarkastaa ettei siellä ole samoja huoneiden numeroja.

```
//Aloitetaan hotellin varauspalvelu
cout << "Tervetuloa hotellin varauspalveluun!" << endl << "-----" << endl;

cout << "Montako huonetta haluat varata (1-" << kaikki_huoneet << "): " << endl;

//Kysytään huoneiden maara joka halutaan varata
cin >> haluttuja_huoneita;
```

Aloitetaan ulossyöttö konsoliruudulle. Pyydetään käyttäjän syöttämään montako huonetta hän haluaa varata. Jottei varaukset ole yli 20 tai alle 1 niin tarkastetaan se koodilla

```
//Tarkistetaan ettei numerot ole alle 1 tai yli 20
if (haluttuja_huoneita < 1 || haluttuja_huoneita > kaikki_huoneet) {
    cout << "Numero ei ole valilla 1-20! Yrita uudelleen!" << endl;
    return 1;
}
```

Jos numero menee yli 20 tai alle 1 niin ohjelma ilmoittaa virheestä ja sulkee itsensä.

```
//Kysytään kuinka monta yötä halutaan viettää
cout << "Kuinka monta yötä haluat viettää: " << endl;
cin >> yot;

//Tallennetaan henkilön nimi
cout << "Antakaa vielä varaajan etunimi ja sukunimi: " << endl;

cin.ignore();
getline(cin, nimi);
```

Pyydetään öiden määrä jotta saadaan laskettua loppuhinta. cin.ignore puhdistaa cin tilan jotta voidaan tallentaa henkilön nimi string muuttuun getline avulla.

```
//Muodostetaan vektori jossa on henkilölle arvotut huoneet
vector<int> hotelli_huone;

//Arvotaan henkilölle varaukset huoneista
for (int i = 0; i < haluttuja_huoneita; i++) {
    int random_vapaa = rand() % 70 + 1;

    //Tarkastetaan ettei ole varatuissa huoneissa
    if (find(varatut_huoneet.begin(), varatut_huoneet.end(), random_vapaa) == varatut_huoneet.end()) {

        //Tarkastetaan ettei tule tuplavarauksia
        if (find(hotelli_huone.begin(), hotelli_huone.end(), random_vapaa) == hotelli_huone.end()) {
            hotelli_huone.push_back(random_vapaa);
        }
    }
}
```

Muodostetaan vektori hotellihuoneille jotka arvotaan käyttäjälle. Ensin find:in avulla tarkastetaan ettei arvotut käyttäjän huoneet ole samoja kuin varatuissa huoneissa olevat. Sen jälkeen tarkastellaan ettei varata samaa huonetta useampaan kertaan ja lisätään lopuksi vektoriin: hotelli\_huone.

```
//Kerrotaan yhden yön hinta.
int yhdYonH = hinta();
cout << "Yhden yön hinta on: " << yhdYonH << " euroa!" << endl;

//Lasketan yhteis hinta vartuille huoneille.
int kokHinta = yhdYonH * yot;

//Tulostetaan varaustiedot
cout << "Seuraavat huoneet ovat nyt varattuja: ";
for (int huone : hotelli_huone) {
    cout << huone << " ";
}
}
```

Kerrotaan loppuun henkilölle paljonko yksi yö maksaa eli tuodaan aliohjelma joka alussa näytettiin. Lasketaan paljonko yhteishinta tulee olemaan laskemalla aliohjelman yhden yön hinta ja kerrotaan se kuinkamonta yötä halutaan viettää.

Tulostetaan hotellihuoneet jotka ovat varattuja. Koodipätkä (huone : hotelli\_huone) tarkoittaa että se tulostaa jokaisen huoneen yksikerrallaan niin kauan kun numeroita vektorissa on.

```
//Kerrotaan mille nimelle varaus on tehty
cout << endl << "Varaukset on tehty nimelle: " << nimi << endl;

//Kerrotaan henkilölle kuinka paljon yhteishinta tulee olemaan
cout << "Hinta varauksille tulee olemaan: " << kokHinta << " euroa!" << endl;
```

Kerrotaan mille nimelle varaukset on tehty. Kerrotaan käyttäjälle paljonko kokonaishinta tulee olemaan lopussa.

```
//Kysytään halutaanko aloittaa työ alusta vai ei
cout << "Haluatko aloittaa alusta?" << endl;
cout << "(K tai E)" << endl;
cin >> valinta;

//Tarkistetaan että valintana on joko K tai E
while (valinta != 'K' && valinta != 'k' && valinta != 'E' && valinta != 'e') {
    cout << "Virheellinen valinta! Valitse joko (K tai E)" << endl;
    cin >> valinta;
}
```

Kysytään käyttäjältä halutaanko aloittaa varauspalvelu uudelleen. Tarkastetaan, ettei henkilö laita muuta kirjainta kuin K,k tai E,e. Jos laitetaan muu kirjain ohjelma pyytää laittamaan uudelleen kirjaimen.

```
//Jos valinta on K niin järjestelmä alkaa alusta mutta jos E niin se sulkeutuu
} while (valinta == 'K' || valinta == 'k');

cout << "Kiitos ja mukavaa päivän jatkoa!" << endl;

//Suljetaan ohjelma
return 0;
```

Lopussa jos valinta on K tai k siirrytään alkuun jolloin aloitetaan kokonaan kaikki alusta. Jos valinta on E tai e niin ohjelma kiittää sen käyttämisestä ja sulkee itsensä.