

## Sakari Liesjärvi harjoitustyö

Koodissani käytetään "Varaus" nimistä structia, josta löytyy muuttujat varaajan nimelle, huoneen numerolle, huoneiden kokonaismäärälle, hinnalle, varausnumerolle sekä öiden määrälle. Koodissani on ilmoitettu globaalisti Varaus tyyppinen "varaukset" taulukko, johon varauksien tiedot tallennetaan, ja laskuri, joka pitää kirjaa varauksien määrästä. Laskuria käytetään muun muassa tietojen tallentamiseen eri taulukon paikoille ja sen avulla voidaan laskea myös vapaana olevien huoneiden määrä. Laskurin muuttujatyyppi on static int, jotta laskurin arvo säilyisi myös aliohjelmien välillä siirtymisen jälkeen.

Koodin lukeminen alkaa mainista, jossa aluksi arvotaan huoneiden kokonaismäärä väliltä 30–70 ja hinta per yö väliltä 80–100 euroa. Tämän jälkeen saadut arvot sijoitetaan pysyviin int muuttujiin. Aloitusvalikossa on mahdollista tehdä varaus, tarkastella varauksia sekä tarkistaa vapaana olevien huoneiden määrä. Varauksien tekeminen ja tarkastelu tapahtuvat omissa aliohjelmissaan.

Varauksien tekemiseen käytettävässä aliohjelmassa tarkastetaan aluksi global laskurin avulla, että varausten määrä ei ylitä saatavilla olevien huoneiden määrää. Tämän jälkeen arvotaan kyseiselle varaukselle ominainen huoneen numero sekä varausnumero väliltä 10000-99999, minkä lisäksi myös testataan, että samat numerot eivät voi toistua varauksissa kahdesti. Duplikaattien testaamiseen käytetään for-looppia. Käyttäjälle ilmoitetaan varauksen tiedot ja pyydetään nimi ja öiden määrä. Varauksen lopuksi käyttäjältä kysytään, että haluaako tämä jatkaa varausten tekemistä vai siirtyä takaisin aloitusvalikkoon ja näytetään myös yhteenveto tehdystä varauksesta. Samassa yhteydessä tehty varaus tallennetaan taulukkoon global laskurin ilmoittamalle paikalle ja laskurin arvoa kasvatetaan 1:llä.

Varauksien tarkasteluun hyödynnettävässä aliohjelmassa on mahdollisuus etsiä varauksia varausnumerolla, nimellä sekä huoneen numerolla. Aliohjelmassa on käytössä oma laskuri, jota käytetään varausten etsimisessä ja vertailussa käyttäjän syöttämiin tietoihin. Koko aliohjelmassa on yksi iso do-while loop sekä pienempiä while looppeja eri hakuvaihtoehtoissa. Olisin käyttänyt mielelläni for-looppeja mutta sain ohjelman toimimaan paremmin whilen avulla. Jokainen hakuvaihtoehto toimii suunnilleen samalla periaatteella, eli while loopilla vertaillaan varausten tietoja käyttäjän syötteeseen ja if-lauseiden avulla katetaan kaikki mahdolliset tapaukset.

Syötteen tarkistus toimii menuissa switchin avulla. Päätin käyttää käyttäjän syötteisiin char muuttujatyyppiä, koska sen avulla voidaan estää helposti kaikkiin ylimääräisiin merkkeihin liittyvät ongelmatapaukset. Int tyyppisissä syötteissä, kuten huoneen numeroissa ja öiden määrässä käytin kuitenkin boolia ja do-while looppia syötteen tarkastukseen. Ainoa merkittävästi puutteellinen asia löytyy varaajan nimeä syötettäessä. Ilman setlocale -komentoa getline osaa ottaa ääkköset oikein mutta tulosteet eivät tietenkään toimi

halutulla tavalla. Setlocalen kanssa ohjelma muuttaa toimivat ääkköset syötteessä vääriksi merkeiksi.

Koen, että työni ylittää helposti arvosanaan 3 mutta yritin myös lisäillä hieman ylimääräisiä ominaisuuksia.