

Harjoitustyö raportti

Tässä raportissa käyn läpi c++ ohjelmoinnin perusteiden lopputyötä jonka toteutuksena oli hotellinvarausrakennusjärjestelmä. Lähdin projektiin tavoitellen parasta arvosanaa ja aloin heti tekemään 5 pisteen työtä. Uskon että se oli oikea valinta, koska pystyin tekemään selkeän suunnitelman projektista sekä ohjelmaan tarvittavista toiminnosta.

1 Projektin työosuus

Aloitin projektin tekemällä pääfunktioon varausjärjestelmän päävalikon ja lisäsiihen vaihtoehdot ominaisuuksille mitä ohjelmaan halusin. Myöhemmin siirsin päävalikon kumminkin omaan aliohjelmaan, jotta pääfunktio pysyisi mahdollisimman siistinä.

Jotta pystyin rakentamaan nämä ominaisuudet, minun piti ensin luoda hotellille pohja. Tein funktion, joka arpoo huoneiden määrän väliltä 40–300 ja varmistaa että luku on parillinen. Rakensiin huoneille rakenteen (Huonenumero, sänkyjen määrä, vapaa/varattu, varajan nimi, öiden määrä, hinta) ja loin sen avulla vektorin, johon tallentuu aiemmin määritelty määrä huoneita. Nämä tiedot oli tietysti tallennettava tekstitiedostoon, jotta uudelleen käynnistäessä varaukset eivät katoa. Tein simpelien funktion vektorin tiedostoon tallentamiselle. Lisäsin pääohjelmaan ominaisuuden, joka tarkistaa onko tiedosto olemassa vai ei joko luo tai lataa datan tiedostosta.

Tämän jälkeen kirjoitin ohjelman perusominaisuudet. Lopuksi lisäsiihin vielä mm. syötteentarkistus aliohjelman sekä siistin käyttöliittymää. Ajanpuutteen vuoksi jää yksi ominaisuus jota olisin kaivannut; hakufunktion toiminta kirjainkoosta huolimatta.

2 Ohjelman toiminta

Ohjelmassa on hyödynnetty aliohjelmaa, joka tyhjentää konsolin näkymän. Aliohjelma on toteutettu siten että se käyttää käyttäjän käyttöjärjestelmän mukaan oikeaa komentoa tähän, välttäen virheet. Konsolin tyhjentäminen helpottaa käyttöliittymän selkeydessä sekä parantaa käyttökokemusta.

Ohjelman alussa on määritelty käytettävät kirjastot, huoneiden rakenne, vakiohinnat/yö ja aliohjelmien prototyypit.

Itse ohjelma alkaa pääfunktiosista. Ensimmäisenä on srank alustus koska eri vaiheissa ohjelmaa käytetään sitä, sitten luodaan vektori. Tämän jälkeen ohjelma tarkistaa onko roominfo.txt tiedosto olemassa. Jos tiedosto löytyy, ohjelma siirtyy aliohjelmaan joka kopio tiedot aiemmin luotuun vektoriin. Jos tiedostoa ei ole olemassa, ohjelma siirtyy aliohjelmaan, joka alustaa hotellin. Tätä samaa ohjelmaa käytetään myös, jos käyttäjä valitsee valikosta järjestelmän resetoimisen.

Kun ohjelman on suorittanut yllä mainitut toiminnot, se siirtyy päävalikko aliohjelmaan. Tässä käyttäjä voi valita mitä haluaa tehdä. Vaihtoehtoja ovat: Varauksen haku, varauksen luonti, varauksen peruminen, järjestelmän resetointi, ohjelman poistuminen. Ohjelman navigoidaan syöttämällä numeroita, ja kaikki syötteet tarkistetaan aina omalla aliohjelmalla, jolle annetaan syöteen minimi/maksimi luvut.

Huomion arvoista mainita, että missä vaiheessa tahansa käyttäjä voi syöttää "0", ja ohjelma palaa päävalikkoon. Tämä kerrotaan myös käyttäjälle.

Jos käyttäjä valitsee päävalikossa hakea varauksia, ohjelma siirtyy hakufunktioon. Ensiksi ohjelma tarkistaa, onko yhtäkään varausta ylipäättääseen tehty. Jos varauksia ei löydy, käyttäjää pyydetään joko tekemään varaus, tai poistumaan päävalikkoon. Jos käyttäjä valitsee tehdä varauksen, ohjelma siirtyy varausfunktioon. Funktio on sama, johon ohjelma siirtyisi, jos käyttäjä siirtyisi päävalikosta tekemään varausta. Jos ohjelma löytää varauksia, voi käyttäjä etsiä niitä joko varausnumerolla tai varaaajan nimellä. Haun onnistuessa ohjelma käyttää aliohjelmaa, joka on puhtaasti varauksen tietojen tulostamiseen. Tämän jälkeen käyttäjä voi valita, haluaako tehdä uuden haun vai palata päävalikkoon.

Jos taas käyttäjä valitsee päävalikosta varauksen tekemisen, ohjelma siirtyy luontaisesti varausfunktioon. Ohjelma huoneiden kokonaismäärän sekä vapaiden 1 ja 2 sängyn huoneiden määrän. Käyttäjää pyydetään syöttämään varaaajan nimi, sitten sänkyjen määrä ja lopuksi öiden määrä. Tässä vaiheessa ohjelma arpoo varausnumeron. Tässä on hyödynnetty aliohjelmaa, joka tarkistaa onko varausnumero jo käytössä, milloin ohjelma pyörittää numeron arvontaa, kunnes uniikki numero löytyy. Ohjelma arpoo myös alennuksen (0 %, 10 % tai 20 %). Käyttäjälle kerrotaan varauksen tiedot ja kysytään, halutaanko varaus vahvistaa. Jos varaus vahvistetaan, ohjelma kirjoittaa sen vektoriin ja sen jälkeen tallentaa tiedostoon. Käyttäjältä kysytään, haluaako hän tehdä uuden varauksen vai palata päävalikkoon.

Päävalikosta siirryttääessa varauksen perumiseen, ohjelma siirtyy taas omaan aliohjelmaansa. Käyttäjältä pyydetään varausnumeroa. Jos varausnumerolla löytyy varaus, ohjelma tulostaa sen tiedot ja pyytää varmistusta perumiseen. Jos käyttäjä päättää perua varauksen, kyseinen huone alustetaan vektorissa sekä tiedostossa takaisin tyhjäksi. Ohjelma kysyy käyttäjältä, haluaako hän perua toisen varauksen vai palata päävalikkoon.

Viimeisenä päävalikosta voi alustaa hotellin. Se yksinkertaisesti käyttää samaa aliohjelmaa mitä käytettäisiin, jos hotellin tietoja ei ohjelmaa käynnistäessä löytyisi.

3 Pohdinta

Opin projektista erittäin paljon. Sen lisäksi että pystyn nyt paljon sulavammin lukemaan koodia ja ymmärtämään sitä, looginen ajatteluni siitä miten ohjelma käyttäätyy, kehittyi huomattavasti. Alussa minulla oli enemmän vaikeuksia ymmärtää, että miksi jokin asia toimii tietyllä tavalla, mutta loppua kohden se vaati vain pienen vilkaisun koodiin. Tässä on silti tieteenkin vielä paljon opittavaa.