Tehtävä 4: Readers & Writers -ongelma stadionkonsertin lipunvaraustietojärjestelmässä

Readers & Writers -ongelmaa voidaan soveltaa stadionkonsertin lipunvaraustietojärjestelmään varmistaaksemme, että useita ihmisiä voi tarkastella ja varata lippuja samanaikaisesti, mutta vain yksi voi tehdä muutoksia (kirjoittaa) kerrallaan. Tämä on tärkeää, jotta varausjärjestelmä toimii oikein ja estää ristiriitoja lippujen varaamisen aikana.

Idea:

1. Lukijat (readers) voivat olla kaikki stadionkonsertin asiakkaat, jotka selaavat ja tarkastelevat lippuja sekä katsomopaikkoja.

2. Kirjoittaja (writer) voi olla järjestelmän ylläpitäjä tai asiakas, joka tekee muutoksia lipunvarausjärjestelmään, kuten lisäämällä uusia lippuja myyntiin tai päivittämällä lippujen tietoja.

Ratkaisu:

Käytetään Readers & Writers -algoritmia varmistamaan samanaikainen luku- ja kirjoitusmahdollisuus stadionkonsertin lipunvaraustietojärjestelmässä:

1. Käytä kahden semaforin lähestymistapaa: yksi lukijoille (readers) ja toinen kirjoittajalle (writer). Alustetaan seuraavasti:

```c

semaphore read\_mutex = 1; // Lukijoiden lukusemafori

semaphore write\_mutex = 1; // Kirjoittajan lukusemafori

int read\_count = 0; // Lukijoiden lukumäärä

```

2. Lukijan (reader) on saadakseen pääsyn tietokantaan:

```c

down(&read\_mutex); // Lukijoiden lukusemafori lukitsee itsensä

read\_count = read\_count + 1; // Lisätään yksi lukija

if (read\_count == 1) {

down(&write\_mutex); // Kirjoittajan lukusemafori lukitaan, jos tämä on ensimmäinen lukija

}

up(&read\_mutex); // Lukijoiden lukusemafori vapautetaan

// Lue tietokantaa

down(&read\_mutex); // Lukijoiden lukusemafori lukitsee itsensä uudelleen

read\_count = read\_count - 1; // Yksi lukija vähemmän

if (read\_count == 0) {

up(&write\_mutex); // Jos tämä oli viimeinen lukija, kirjoittajan lukusemafori vapautetaan

}

up(&read\_mutex); // Lukijoiden lukusemafori vapautetaan

```

3. Kirjoittajan (writer) on saadakseen pääsyn tietokantaan:

```c

down(&write\_mutex); // Kirjoittajan lukusemafori lukitsee itsensä

// Kirjoita tietokantaan

up(&write\_mutex); // Kirjoittajan lukusemafori vapautetaan

```

Tämä ratkaisu mahdollistaa useiden lukijoiden samanaikaisen lukemisen tietokannasta ja varmistaa, että vain yksi kirjoittaja voi tehdä muutoksia tietokantaan kerrallaan. Tämä on tärkeää varmistaakseen lipunvaraustietojärjestelmän eheyden ja estää ristiriitoja varausprosessin aikana.