

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики
Кафедра системного програмування і спеціалізованих комп'ютерних
систем

Лабораторна робота №2

з дисципліни “Введення до оперативних систем”

Тема: “Синхронізація процесів”

Варіант 9

Виконав:

студент IV-го курсу ФПМ

групи КВ-82

Іваненко Олександр Андрійович

Київ 2021

Метою лабораторної роботи є ознайомлення з особливостями взаємодії процесів і засобами їх синхронізації при доступі до спільно використовуваних ресурсів.

Завдання

Об'єкт моделювання: *Автомат для продажу авіабілетів.*

Автомат приймає гроші (тут тільки одного визначеного номіналу – 1грн.) і видає здачу монетами вартістю до 1 грн. (1, 2, 5, 10, 25, 50 коп.). Сума здачі розраховується. Початкова кількість монет кожного номіналу задається і становить:

1 коп. – 50 шт., 2 коп. – 25 шт.,
5 коп. – 20 шт., 10 коп. – 15 шт.,
25 коп. – 10 шт., 50 коп. – 5 шт.,

Введення запиту на продаж здійснюється шляхом вибору певного пункту меню:

0 – включити автомат; 1 – купити білет до Києва вартістю 28 коп.;
2 – білет до Москви вартістю 37 коп.; 3 –білет до Лондона вартістю 50 коп.;
4 – білет до Берліна вартістю 77 коп.; 5 –білет до Парижа вартістю 91 коп.
Якщо здачу видати можливо, програма формує потрібний набір монет для здачі (також коригує банк монет) і формує сигнал на видачу. Якщо потрібних купюр для здачі не достає, формується відповідне повідомлення. Вимоги на видачу грошей надходять після чергового сеансу продажу або відмови.

Кількість терміналів і процесів: Модель автомата представити у вигляді *двох* взаємодіючих процесів А і В. Процес А визначає факти надходження вимог на продаж і потрібну суму здачі. Процес В очікує момент появи необхідності видати здачу і, якщо величина здачі відома, визначає кількість і номінали потрібних монет або неможливість видачі.

Засоби синхронізації: Для організації доступу до подільних ресурсів використати *поштові скриньки*.

Програмний код

```
package com;

import java.util.Arrays;
import java.util.List;
import java.util.Scanner;
import java.util.stream.Collectors;

class Coin {
    int size;
    int amount;
    Coin(int size, int amount){
        this.size = size;
        this.amount = amount;
    }
}

class Bank {
    static Coin [] bankCoins = {
        new Coin(50, 5),
        new Coin(25, 10),
        new Coin(10, 15),
        new Coin(5, 20),
        new Coin(2, 25),
        new Coin(1, 50),
    };

    static List<Coin> needCoins = Arrays.stream(bankCoins).map( coin -> new
Coin(coin.size, 0)).collect(Collectors.toList());

    static void printBank(){
        System.out.println("Це всі наші гроші ДИВІТЬСЯ :");
        for(Coin coin : bankCoins){
            System.out.printf("%3d $ - %3d coins\n", coin.size, coin.amount);
        }
    }

    static boolean getChange(int n) {
        for (int i = 0; i < bankCoins.length; i++) {
            int need = Math.floorDiv(n, bankCoins[i].size);
            if (bankCoins[i].amount - need < 0)
                needCoins.get(i).amount = bankCoins[i].amount;
            else needCoins.get(i).amount = need;
            n -= bankCoins[i].size * needCoins.get(i).amount;
        }
        if(n != 0) return false;
        for (int i = 0; i < bankCoins.length; i++){
            bankCoins[i].amount -= needCoins.get(i).amount;
        }
        return true;
    }
}
```

```

class PostBox{
    static boolean sent = false;
    static int change = -1;
    static int signal = 0;
}

class ThreadA extends Thread{
    @Override
    public void run() {
        Scanner sc = new Scanner(System.in);
        System.out.println("\n\n+-----+
+-----+");
        System.out.println("|    Привіт! Ви можете купити квиток на літак
(Ящко не хворієте COVID)!    |");
        System.out.println("+-----+
+-----+\n");
        System.out.print(" Щоб увімкнути автомат натисніть '0': ");
        PostBox.signal = sc.nextInt();
        while(PostBox.signal != 9) {
            if (!PostBox.sent) {
                System.out.println("+-----+
+-----+");
                System.out.println("|    Вимкнути автомат.
Натисніть '9' |");
                System.out.println("|    Білет до Києва.    Вартість 28 коп.
Натисніть '1' |");
                System.out.println("|    Білет до Москви.    Вартість 37 коп.
Натисніть '2' |");
                System.out.println("|    Білет до Лондона. Вартість 50 коп.
Натисніть '3' |");
                System.out.println("|    Білет до Берліна. Вартість 77 коп.
Натисніть '4' |");
                System.out.println("|    Білет до Парижа.    Вартість 91 коп.
Натисніть '5' |");
                System.out.println("+-----+
+-----+");
                System.out.print("    Оберіть напрямок: ");
                PostBox.signal = sc.nextInt();

                if (PostBox.signal == 9) {
                    System.out.println("Гарного настрою та не хворійте!");
                    break;
                }

                System.out.print("Введіть гроші будь ласка: ");
                int money = sc.nextInt();
                int price = -1;
                boolean bad = false;
                switch (PostBox.signal) {
                    case 1:
                        price = 28;
                        break;
                    case 2:
                        price = 37;
                        break;
                    case 3:
                        price = 50;

```

```

        break;
    case 4:
        price = 77;
        break;
    case 5:
        price = 91;
        break;
    default:
        bad = true;
        break;
    }
    if (bad) {
        System.out.println("Невідомий напрямок. ВИБАЧТЕ. Оберіть
ЗНОВ.");
        continue;
    }
    if (money - price < 0) {
        while (money - price < 0) {
            System.out.println("Маловато ви дали. ЩЕ :");
            money += sc.nextInt();
        }
    }
    PostBox.change = money - price;
    PostBox.sent = true;

    }
    try {
        Thread.sleep(1000);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
    }
    Bank.printBank();
}

}

class ThreadB extends Thread{

    @Override
    public void run() {
        while (PostBox.signal != 9) {
            Scanner sc = new Scanner(System.in);
            if (PostBox.sent) {
                if (Bank.getChange(PostBox.change)) {
                    System.out.printf("Ваша решта '%d' та ваш квиток.
Отримайте.\n", PostBox.change);
                } else {
                    System.out.printf("Вибачте. Заберіть гроші. Немає такої
решти (%d коп.) :( ", PostBox.change);
                }
                System.out.println("Продовжити купування, натисніть '0'.
Вимкнути апарат, натисніть '9'.");
                PostBox.signal = sc.nextInt();
                PostBox.sent = false;
            }
        }
    }
}

```

```
        Thread.sleep(1000);
    } catch (InterruptedException e) {
        e.printStackTrace();
    }
}

}

}

public class Main {
    public static void main(String[] args) {
        ThreadA a = new ThreadA();
        ThreadB b = new ThreadB();
        a.start();
        b.start();
    }
}
```

Результат

```
+-----+
|   Привіт! Ви можете купити квиток на літак (Ящко не хворієте COVID)!   |
+-----+

Щоб увімкнути автомат натисніть '0': 0
+-----+
|  Вимкнути автомат.                                Натисніть '9'  |
|  Білет до Києва.   Вартість 28 коп.   Натисніть '1'  |
|  Білет до Москви.  Вартість 37 коп.   Натисніть '2'  |
|  Білет до Лондона. Вартість 50 коп.   Натисніть '3'  |
|  Білет до Берліна. Вартість 77 коп.   Натисніть '4'  |
|  Білет до Парижа.  Вартість 91 коп.   Натисніть '5'  |
+-----+

Оберіть напрямок: 1
Введіть гроші будь ласка: 150
Ваша решта '122' та ваш квиток. Отримайте.
Продовжити купування, натисніть '0'. Вимкнути апарат, натисніть '9'.
0
+-----+
|  Вимкнути автомат.                                Натисніть '9'  |
|  Білет до Києва.   Вартість 28 коп.   Натисніть '1'  |
|  Білет до Москви.  Вартість 37 коп.   Натисніть '2'  |
|  Білет до Лондона. Вартість 50 коп.   Натисніть '3'  |
|  Білет до Берліна. Вартість 77 коп.   Натисніть '4'  |
|  Білет до Парижа.  Вартість 91 коп.   Натисніть '5'  |
+-----+

Оберіть напрямок: 4
Введіть гроші будь ласка: 140
Ваша решта '63' та ваш квиток. Отримайте.
Продовжити купування, натисніть '0'. Вимкнути апарат, натисніть '9'.
9
Це всі наші гроші ДИВІТЬСЯ :
50 $ - 2 coins
25 $ - 10 coins
10 $ - 12 coins
5 $ - 20 coins
2 $ - 23 coins
1 $ - 49 coins

Process finished with exit code 0
```

Висновки

Була виконана робота по синхронізації процесів. У нашій роботі існує два потоки. Один виконує прийом грошей та підрахування решти. Інший видає решту, або повідомлення о неможливості.

Засобом синхронізації є поштова скринька. У нашій роботі це структура з певними полями, через яку два процеси обмінюються повідомленнями. Наприклад процес А коли підрахував значення, відправляє повідомлення, що він виконав свою частину, тоді включається процес Б, і виконує свою частину, відправляючи повідомлення.

Я вважаю, що поштова скринька досить зручний спосіб, але не дуже надійний та популярний. Наприклад семафори монітори та м'ютекси існують в реалізаціях багатьох мов, та за допомогою них можна створювати більш надійні способи синхронізації. Оскільки поштова скринька – просто структура даних, звичайний програміст може щось упустити. А в багатопотоковому виконанні, знайти помилки в реалізації дуже важко.