Міністерство освіти та науки України

Національний технічний університет України “КПІ”

Факультет інформатики та обчислювальної техніки

Кафедра автоматизованих систем обробки інформації і управління

**ЗВІТ**

про виконання

комп’ютерного практикуму №2

на тему:

**«Дослідження основних принципі побудови імітаційних моделей мовою GPSS»**

**Завдання 13**

**Варіант (5) 2**

**Виконав:** студент групи ІС-32

Капорін Роман

Київ 2016

**Мета роботи:**

Дослідження можливостей і придбання навичок використання імітаційних моделей для отримання ймовірнісних часових характеристик процесу функціонування досліджуваного об'єкту і знаходження оптимальних стратегій управління ним.

1. **Постановка задачі**

*Завдання*. Невеликий продовольчий магазин складається з трьох прилавків і декількох кас на виході з магазину. Покупці приходять в магазин, вхідний потік має пуасонівський характер, причому середнє значення інтервалу часу приходу складає **Тпр** (числові дані по варіантах дивись у *табл. 2.2*). Якщо, увійшовши до магазину, покупець бачить, що черга на вихід з магазину більше, ніж **L** людина, він розгортається і йде. Інакше покупець бере кошик і може обійти один або декілька прилавків, відбираючи продукти. Ймовірність обходу конкретного прилавка показана в *табл. 2.3*. Час, потрібний для обходу прилавка та число покупок, вибраних біля прилавка, розподілені рівномірно. Детальна інформація по кожному з прилавків приведена в *табл. 2.3*.

Після того, як товар відібраний, покупець стає в кінець загальної черги до кас. Вже стоячи в черзі, покупець може захотіти зробити ще **N ± 1** покупок. Час обслуговування покупця касиром пропорційний числу зроблених покупок, на одну купівлю йде **Т** с .

Підраховано, що кожен "втрачений" покупець обходиться магазину в **С** одиниць вартості. На оплату одного касира витрачається **Zпл** одиниці вартості в день. Час роботи магазину складає **10** годин.

Використовуючи модель, що описує процес покупок в продуктовому магазині, визначити таку кількість касирів, при якому досягалася б мінімальна величина витрат. Витрати включають заробітну плату касирів і втрати від зменшення кількості покупців, що вирішили зробити покупки в іншому магазині. Вважаючи, що число кошиків необмежено, визначити яку максимальну кількість кошиків потрібно мати у магазині.

Таблиця 2.2 - Числові значення характеристик системи

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Варіант |  | L | N | T | C | Z |
| 5 | 18 | 7 | 1 | 2.5 | 0.15 | 35 |

Таблиця 2.3 - Характеристика покупок у прилавків

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Варіант | Номер прилавка | Ймовірність покупки | Час обходу прилавка | Число покупок |
| 5 | 1 | 0.68 | 250±50 | 4 |
| 2 | 0.75 | 350±80 | 6±2 |
| 3 | 0.65 | 200±40 | 5±3 |

1. **Структурна схема**

**C:\Users\Roman Kaporin\Desktop\LAB_13\LAB_13.png**

Рис. 1 – Структурна схема моделі

1. **Розрахунок варіантів кількості касирів**

Сумарна середня кількість покупок у одного покупця – сума добутків ймовірностей обходу кожного з прилавків на відповідне середнє значення покупок, а також середнє число додаткових покупок

Середній час обслуговування покупця на касі – добуток сумарного середнього числа покупок на час обробки однієї покупки:

При цьому середнє значення часу надходження покупців дорівнює 18 секунд, що не перевищує середнє значення обслуговування покупця.

З цього можна зробити висновок, що достатньо 3-х касирів для нормального функціонування магазину.

1. **План імітаційних експериментів**

Виходячи з розрахунків, імітаційні експерименти почнемо для кількості касирів, рівного одиниці. Потім будемо збільшувати число касирів на 1 на кожному кроці, проводячи прогони моделі для 5 різних випадків по 5 прогонів кожний.

1. **Лістинг програми**

PUASON\_DISP FUNCTION RN1,C24

0,0/.1,.104/.2,.222/.3,.355/.4,.509/.5,.69/.6,.915

.7,1.2/.75,1.38/.8,1.6/.84,1.83/.88,2.12/.9,2.3

.92,2.52/.94,2.81/.95,2.99/.96,3.2/.97,3.5/.98,3.9

.99,4.6/.995,5.3/.998,6.2/.999,7/.9998,8

CUSTOMER\_INCOME\_TIME VARIABLE 18 ;Час попадання покупців в магазин

MAX\_QUEUE\_SIZE VARIABLE 7 ;Максимальна черга в магазинi

PRODUCT\_PRICE VARIABLE 0.15 ; C

CASHIER\_SALLARY VARIABLE 35 ; Z

PROBABILITY\_STORE\_1 VARIABLE 0.68

PROBABILITY\_STORE\_2 VARIABLE 0.75

PROBABILITY\_STORE\_3 VARIABLE 0.65

INITIAL X$TIME\_ADVANCE,0

INITIAL X$COSTS,0

ADD\_STAFF FUNCTION RN1,C2 ; число додаткових покупок (1)

0,0/1,2 ; N+-1 (N=1)

AYL1 FUNCTION RN1,C2 ; число покупок у 1го прилавка

0,4/1,4 ; 4

AYL2 FUNCTION RN1,C2 ; число покупок у 2го прилавка

0,4/1,8 ; 6+-2

AYL3 FUNCTION RN1,C2 ; число покупок у 3го прилавка

0,2/1,8 ; 5+-3

CASHIER STORAGE 3 ; кількість касирів

BASKETS STORAGE 1000000 ; кількість кошиків

GENERATE V$CUSTOMER\_INCOME\_TIME,FN$PUASON\_DISP

TEST LE Q$QCASHIER,V$MAX\_QUEUE\_SIZE,WAS\_NOT\_ENTERED ; якщо черга завелика, відмова в обслуговуванні

ENTER BASKETS ; взяти кошик

ASSIGN PRODUCTS\_COUNTER,0 ; кількість покупок

TRANSFER V$PROBABILITY\_STORE\_1,ENTER\_STORE\_2,STORE\_1 ; зробити покупку у прилавка 1?

STORE\_1 ADVANCE 250,50 ; затримка у прилавка 1

ASSIGN PRODUCTS\_COUNTER+,FN$AYL1 ; взяти покупки з прилавка 1

ENTER\_STORE\_2 TRANSFER V$PROBABILITY\_STORE\_2,ENTER\_STORE\_3,STORE\_2 ; зробити покупку у прилавка 2?

STORE\_2 ADVANCE 350,80 ; затримка у прилавка 2

ASSIGN PRODUCTS\_COUNTER+,FN$AYL2 ; взяти покупки з прилавка 2

ENTER\_STORE\_3 TRANSFER V$PROBABILITY\_STORE\_3,OUT,STORE\_3 ; зробити покупку у прилавка 3?

STORE\_3 ADVANCE 200,40 ; затримка у прилавка 3

ASSIGN PRODUCTS\_COUNTER+,FN$AYL3 ; взяти покупки з прилавка 3

OUT QUEUE QCASHIER ; зайняти чергу в касу

ASSIGN PRODUCTS\_COUNTER+,FN$ADD\_STAFF ; додаткова покупка

ENTER CASHIER ; зайняти касира

DEPART QCASHIER ; покинути чергу в касу

SAVEVALUE TIME\_ADVANCE,(P$PRODUCTS\_COUNTER#3.3)

ADVANCE X$TIME\_ADVANCE ; оформлення касиром покупки

LEAVE CASHIER ; звільнити касира

LEAVE BASKETS ; повернути кошик

TRANSFER ,OUTH ; йти до виходу

WAS\_NOT\_ENTERED ADVANCE 0 ; підрахунок покупців без покупок

OUTH TERMINATE 0

GENERATE (3600#10) ; моделювання 10 годин роботи (10 год. = 36000 с)

SAVEVALUE COSTS,(N$WAS\_NOT\_ENTERED#V$PRODUCT\_PRICE+(R$CASHIER+S$CASHIER)#V$CASHIER\_SALLARY) ;кількість втрачених покупців на вртість втрати + зарплата всім касирам

SAVEVALUE VAR\_BACKETS,(SM$BASKETS) ; максимальна кількість занятих кошиків

TERMINATE 1

1. **Аналіз результатів**

|  |
| --- |
| **C:\Users\Roman Kaporin\Desktop\ANOVA(RES,2,1).PNG** |
| Рис. 2 - Фінальний результат моделювання для кількості касирів |
| **C:\Users\Roman Kaporin\Desktop\ANOVA(BACKETS,2,1).PNG** |
| Рис. 3 - Фінальний результат моделювання для кількості кошиків |

З ростом числа касирів затрати тільки ростуть і оптимальним є 3 касири. Перевіримо кількість необхідних кошиків. Максимальна кількість занятих кошиків досягається при трьох касирах та дорівнює 57. При збільшенні кількості касирів кількість кошиків знижується до 53 і лишається незмінною

1. **Висновок**

Для заданої моделі було встановлено, що оптимальна кількість касирів з точки зору мінімізації затрат – 3, а максимальна кількість одночасно зайнятих кошиків – 57 досягається при 3-х касирах. При збільшенні числа касирів кількість кошиків падає до 53 і не змінюється більше.

1. **Відповіді на контрольні запитання**
2. Ми використовуємо БКП на корзини, щоб побачити найбільшу кількість корзин, що можуть використовувати користувачі одночасно. Тому ми не обмежуємо БКП зверху
3. Ми використовуємо БКП в якості касирів. Таким чином формуються черги до кас, якщо всі зайняті. Для моніторингу черги використовуємо Queue OCashier.
4. При входженні в магазин ми зберігаємо кількість куплених продуктів в параметрі транзакта (користувача) countProducts.
5. Він зберігає значення параметра PRODUCTS\_COUNTER.
6. Блок Advance 0 моделює взаємодію покупця з магазином. Ми використовуємо його для того щоб вкінці порахувати кількість втрачених покупців через N$WAS\_NOT\_ENTERED. При Terminate 0 ми не отримаємо правильних данних СЧА. Оскільки використання Advance не є правильним, збоку цільового призначення цієї команди, ми можемо альтернативно замінити цю команду на SAVEVALUE по-елементно додаючи кількість незадоволених покупців.
7. Змінити BASKETS Storage 100000 на 75.
8. Можливість = 0. (Див код програми).