

Scientific journal  
**PHYSICAL AND MATHEMATICAL EDUCATION**  
Has been issued since 2013.

ISSN 2413-158X (online)  
ISSN 2413-1571 (print)

Науковий журнал  
**ФІЗИКО-МАТЕМАТИЧНА ОСВІТА**  
Видається з 2013.



<http://fmo-journal.fizmatsspu.sumy.ua/>

*Федусенко О.В. Використання програмного модуля імітаційного моделювання роботи лікарняної палати при вивченні систем масового обслуговування. Фізико-математична освіта. 2018. Випуск 1(15). С. 327-331.*

*Fedusenko O. Use Of The Program Module Of The Immated Modeling Of The Work Of The Hospital Chamber In The Studies Of Mass-Service Systems. Physical and Mathematical Education. 2018. Issue 1(15). P. 327-331.*

УДК 519.876.5

**О.В. Федусенко**

Київський національний університет імені Тараса Шевченка, Україна

evfedus@gmail.com

DOI 10.31110/2413-1571-2018-015-1-063

#### **ВИКОРИСТАННЯ ПРОГРАМНОГО МОДУЛЯ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ РОБОТИ ЛІКАРНЯНОЇ ПАЛАТИ ПРИ ВИВЧЕНІ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ**

**Анотація.** Стаття присвячена проблемі вивчення студентами систем масового обслуговування та імітаційного моделювання, оскільки ці теми є досить складними та часто не мають наочних прикладів. В статті запропоновано програмний модуль, який дозволить покроково провести імітаційне моделювання системи масового обслуговування з відмовами. У програмному модулі, що розробляється автором, система масового обслуговування представлена у вигляді лікарняної палати.

У статті розглянуті основні принципи побудови систем масового обслуговування та імітаційного моделювання. Наведено основні етапи імітаційного моделювання. Сформована задача імітаційного моделювання роботи лікарняної палати, як системи масового обслуговування з відмовами, визначені основні функціональні вимоги до програмного модуля та наведено початкові значення величин для моделювання. Головними функціональними вимогами до модуля є можливість проведення покрокового моделювання та його відображення, як у вигляді таблиці, так і у вигляді гістограми, а також можливість читання початкових даних з файлу та збереження отриманих результатів у текстовий файл або експорт їх у Excel. При розробці програмного модуля автором використано сучасну методологію об'єкто-орієнтованого програмування SOLID відповідно до якої і побудовано діаграму класів програмного модуля, що наведена у статті. Використання даної методології дозволяє, в майбутньому, досить легко розширювати та підтримувати подальше функціонування модуля, додаючи до нього як нові функціональні можливості, так і нові модулі.

Автором описано алгоритм роботи модулю відповідно до діаграми класів та надано результати роботи програмного модуля. Результати роботи програмного модуля наведено як в табличному так і в графічному представленні, у вигляді гістограми. Отримані результати, майже повністю, збігаються з результатами наведеними в інших дослідженнях, а похибка знаходиться у допустимому інтервалі.

Розроблений програмний модуль може бути використано студентами під час вивчення дисципліни «Моделювання систем», що дозволить підвищити якість навчання за рахунок використання наочного прикладу моделювання системи масового обслуговування з відмовами.

**Ключові слова:** система масового обслуговування, імітаційне моделювання, об'єкто-орієнтоване програмування, SOLID.

**Постановка проблеми.** Системами масового обслуговування(СМО) називають такі системи, в яких у випадкові моменти часу надходять заявки на обслуговування. При цьому заявки, що надійшли обслуговуються за допомогою наявних у розпорядженні системи каналів обслуговування[1].

Основними компонентами системи масового обслуговування будь-якого виду є [2]:

- Вхідний потік вимог, що поступають або заявок на обслуговування;
- Дисципліна черги;
- Механізм обслуговування.

Розрізняють два основних види систем масового обслуговування[3]:

– Системи з відмовами, в яких заявка, що надійшла в систему в момент, коли всі канали зайняті, отримує відмову і відразу ж покидає чергу;

– Системи з очікуванням (чергою), в яких заявка, що надійшла в момент, коли всі канали обслуговування зайняті, стає в чергу і чекає, поки не звільниться один з каналів. Системи масового обслуговування з очікуванням діляться на системи з обмеженим очікуванням і системи з необмеженим очікуванням.

Як правило для вивчення роботи СМО використовується імітаційне моделювання.

Імітаційне моделювання є відносно новим і динамічним методом дослідження поведінки систем управління. Цей метод полягає в тому, що за допомогою комп'ютера відтворюється поведінка досліджуваної системи управління, а дослідник, керуючи ходом процесу імітації та оглядаючи одержувані результати, робить висновок про її властивості та якості поведінки[2].

Математичною основою машинної імітації є метод Монте-Карло у розвиток якого значний внесок був внесений роботами Н.П. Бусленко, Д.І. Голенко, І.М. Соболя і др.

Імітаційне моделювання – процес конструювання моделі реальної системи і постановки експериментів на цій моделі з метою або зрозуміти поведінку системи, або оцінити, в рамках обмежень, різні стратегії, що забезпечують функціонування цієї системи[4].

Типова послідовність імітаційного моделювання включає наступні етапи[5]:

- концептуальний: розробка концептуальної схеми і підготовка області вихідних даних;
- математичний: розробка математичних моделей та обґрунтування методів моделювання;
- програмний: вибір засобів моделювання і розробка програмних моделей;
- експериментальний: перевірка адекватності і коректування моделей, планування обчислювальних експериментів, безпосередньо моделювання, інтерпретація результатів.

Однією з проблем вивчення розділів дисциплін, що присвячено системам масового обслуговування, є відсутність наочного прикладу використання СМО та імітаційного моделювання їх роботи. Тому, актуальна розробка програмного модуля імітаційного моделювання, який дозволить покроково провести моделювання системи масового обслуговування з відмовами та наочно відобразить отримані результати у вигляді гістограми.

**Мета статті.** Метою даного дослідження є розробка програмного модуля моделювання роботи лікарняної палати, як системи масового обслуговування з відмовами. Для розробки даного модуля використовується мова програмування C#.

**Виклад основного матеріалу.** Задача моделювання роботи лікарняної палати має наступний вигляд, який наведено в роботі [6]. В середньому за день  $n$ , при кількості днів  $N$ , у палату лікарні надходять  $z$  хворих. Здоров'я людини оцінюється за певною шкалою. Кожен хворий проходить тест, результати якого рівномірно розподілені на інтервалі від  $k_1$  до  $k_2$  балів. Коли в палаті немає вільних місць, хворі з оцінкою вище  $k_3$  балів на лікування не приймаються.

$$k_3 = k_2 - a$$

Всього в палаті  $m$  місць. Хворий виписується з палати, коли його оцінка стає вище  $k_4$  балів.

$$k_4 \geq k_2 + a$$

Оцінка хворого змінюється протягом доби на величину, рівномірно розподілену на інтервалі від  $-1$  до  $l_2$  балів. Коли в палаті немає вільних місць, але надходить потенційний хворий з неї виписується хворий, оцінка якого дорівнює або вище  $k_5$  балів.

$$k_5 = k_2 + a$$

Потрібно оцінити такі величини: середній час перебування хворого в палаті, завантаження палати, число відмов в лікуванні, число виписаних достроково.

Отже програмний модуль моделювання лікарняної палати, з використанням системи масового обслуговування, повинен виконувати наступні основні функції:

- введення основних параметрів моделювання та їх збереження;
- можливість проведення покрокового моделювання та його відображення;
- проведення моделювання та збереження результатів у файл;
- перегляд результатів моделювання у графічному вигляді;
- експорт результатів моделювання у MS Excel.

Для написання модулю була обрана мова програмування C# та об'єктно-орієнтований підхід. При написанні модуля максимально будуть використовуватися принципи SOLID – п'ять основних принципів ООП[7]:

- Принцип єдиної відповідальності (Single responsibility). На кожен об'єкт повинен бути покладений один єдиний обов'язок.
- Принцип відкритості / закритості (Open-closed). Програмні сутності повинні бути відкриті для розширення, але закриті для модифікації.
- Принцип підстановки Барбара Лісков (Liskov substitution). Об'єкти в програмі можуть бути замінені їх нащадками без зміни властивостей програми.
- Принцип поділу інтерфейсу (Interface segregation). Багато спеціалізованих інтерфейсів краще, ніж один універсальний
- Принцип інверсії залежностей (Dependency Inversion). Залежності повинні будуватися відносно абстракцій, а не деталей.

Діаграма класів модулю з використанням основних принципів SOLID буде мати наступний вигляд (рис. 1).

Як можна побачити, діаграма класів вміщує два інтерфейси для запису IWrite та читання IRead. Данні інтерфейси реалізовані за допомогою класів WriteF – запис у файл, WriteExcel – експортування у Excel, Read – читання з форми, ReadF – читання з файлу.

Для зберігання основних вхідних параметрів використовується статичний клас Zpas, а для накопичення статистичних даних використовується статичний клас Stat\_all. Для відображення статистичних даних використовується клас Stat\_gist, поля якого вміщують наступні масиви значень: col – масив кількості днів перебування у лікарні, cis – масив кількості хворих, що перебували певний термін у лікарні, ver – масив ймовірностей перебування у лікарні певний термін. Розрахунок ймовірності відбувається за допомогою методу ras\_ver. Побудова гістограми та функції ймовірності відбувається за допомогою методів graf та graf1 відповідно.

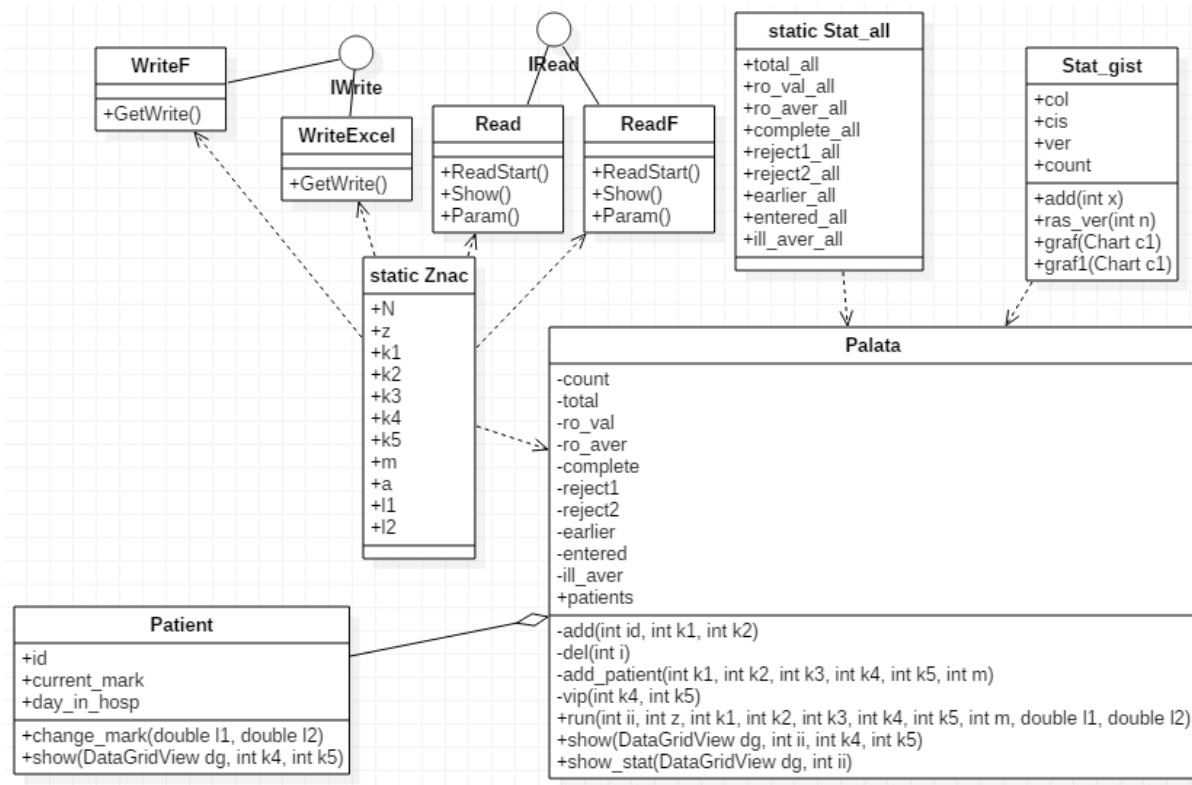


Рис. 1. Діаграма класів

Головним класом модулю є клас Palata, якій у якості поля patients вміщує список пацієнтів. А іншими полями є статистичні значення. Основним методом є метод run який працює за наступним алгоритмом.

1. Перерахування усіх поточних оцінок пацієнтів, які вже лікуються у палаті за допомогою методу change\_mark класу Patient.

2. Знаходження усіх пацієнтів які вже вилікуванні, тобто оцінка стану яких більше або дорівнює k4, та виписка таких пацієнтів.

3. Додавання нових z пацієнтів за допомогою методу add\_patient. Додавання відбувається наступним чином:

3.1 Якщо в палаті є вільне місце, то пацієнт додається у палату.

3.2 Якщо у палаті немає вільного місця та оцінка пацієнта менша за k3, то шукається пацієнт оцінка якого дорівнює або вище за k5.

3.3 Якщо пацієнта знайдено у пункті 3.2 то він виписується за допомогою методу vip, якщо ж такого пацієнта не знайдено, то новий хворий отримує відмову.

4. Збільшення терміну лікування хворих, що лежать у палаті(поле day\_in\_hosp класу Patient).

Розглянемо приклад роботи модуля, введемо наступні вхідні параметри:

- N=1000,
- z=2,
- k1=30,
- k2=44,
- a=3,
- k4=49,
- m=25,
- l1=-0.2,
- l2=1.2.

Головне вікно програмного модуля з вхідними даними та умовами задачі наведено на рис. 2. Вхідні дані можна як ввести з клавіатури, так і зчитати з підготовленого текстового файлу.

Результат покрокового моделювання у вигляді таблиці та гістограми наведено на рис.3. Червоним кольором позначено пацієнтів, яких не можна виписувати, зеленим – пацієнтів які вже закінчили лікування і будуть виписані, а синім кольором – пацієнтів, які не закінчили лікування, але можуть бути виписані.

Розглянемо роботу модуля в цілому. Для цього оберемо пункт «Моделювання» та проведемо 10 експериментів (рис. 4).

Отримані результати майже збігаються з результатами наведеними у роботі Труб І.І.[6], похибка заходиться у допустимому інтервалі.

**Висновки.** Використання модуля імітаційного моделювання роботи лікарняної палати при вивченні розділу СМО у таких дисциплінах, як наприклад, «Моделювання систем», дозволить підвищити якість засвоєння учбового матеріалу, оскільки студенти можуть провести покрокове дослідження роботи імітаційної моделі СМО.

Серед подальших досліджень із цієї тематики доречно запропонувати розробку навчальної інформаційної системи з даного розділу, яка буде вміщувати декілька модулів з різними варіантами СМО.

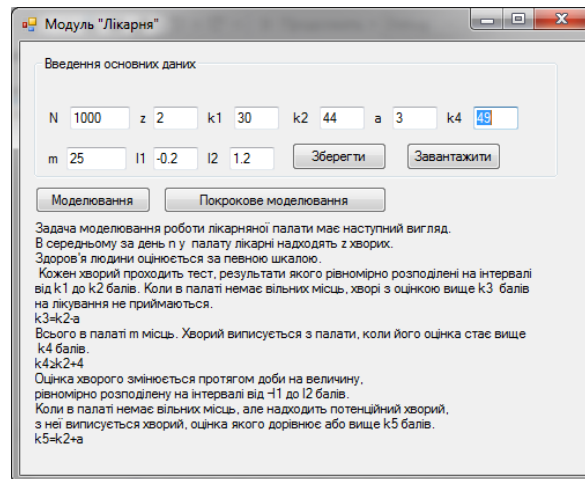


Рис. 2. Головне вікно модулю з основними вхідними параметрами

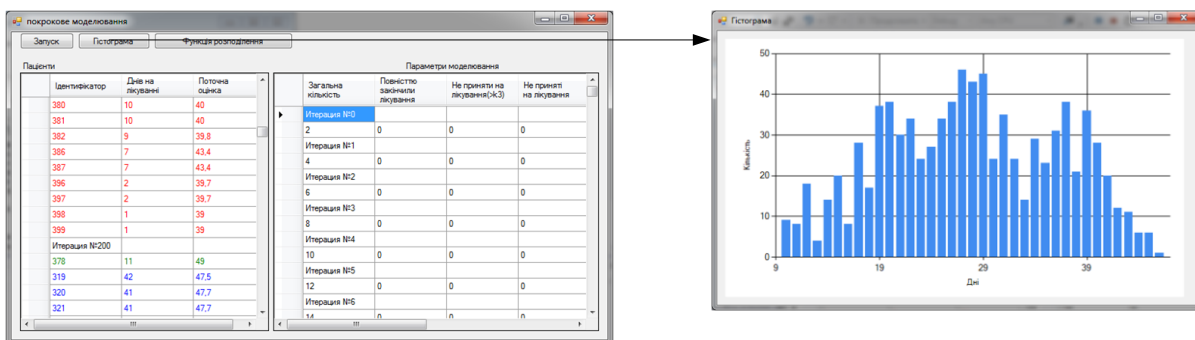


Рис. 3. Результати покрокового моделювання

Моделювання								
Кількість експериментів 10								
	Загальна кількість	Повністю закінчили лікування	Не прийняті на лікування(к3)	Не прийняті на лікування	Виписано доistroво	Всього прийнято	Середнє завантаження	Середній час
2000	837	857	77	210	856	0.88712	16.78102558402	
2000	545	896	5	529	570	0.91304	11.42286238970...	
2000	246	1339	55	335	271	0.972120000000...	51.35770313977...	
2000	911	563	6	495	936	0.879039999999...	10.65101460241...	
2000	393	1055	5	522	418	0.93072	8.745996351508...	
2000	951	758	33	239	970	0.871679999999...	14.12629625523...	
2000	552	867	5	551	577	0.91264	11.64512557234...	
2000	259	1369	56	291	284	0.97228	59.704180811112	
2000	822	603	5	545	847	0.911440000000...	11.01174112142...	
2000	349	1146	7	473	374	0.93588	8.821749287252...	
Всього								
20000	5865	9453	254	4190	6103	0.918596	20.42676950891...	

Рис. 4. Результати моделювання

**Список використаних джерел**

1. Большаков А.С. Моделирование в менеджменте. Учебное пособие. М.: Информационно-издательский дом «Филинь», Рилант, 2000. 464 с.
2. Таха Х. Введение в исследование операций. М.: Издательский дом "Вильямс", 2005. 912 с.
3. Томашевський В.М. Моделювання систем. К.: Видавнична група ВНУ, 2007. 352с.
4. Хусаїнов Д.Я. Введення в моделювання динамічних систем: Навч. посібник/ Д.Я. Хусаїнов, І.І. Харченко, А.В. Шатирко. К.: Київський національний університет імені Тараса Шевченка, 2010. 162 с.
5. Аристов С.А. Имитационное моделирование экономических систем: Учеб. пособие. Екатеринбург: Урал.гос.экон.ун-та. 2004. 123 с.
6. Труб И.И. Объектно-ориентированное моделирование на C++. Учебный курс/ И.И.Труб. СПб.: Питер, 2006. 411 с.
7. Мартин Р., Мартин М. Принципы, паттерны и методики гибкой разработки на языке C#/ Р. Мартин, М. Мартин. Символ-Плюс, 2011. 768 с.

## References

1. Bolshakov A.S. Modeling in management. Textbook / A.C. Bolshakov. - M.: Information-publishing house "Filin", Rilant, 2000. – 464p.
2. Taha H. Introduction to the study of operations / H. Taha. M.: Publishing house "Williams", 2005. – 912p.
3. Tomashevsky V.M. Modeluvannya systems / V.M. Tomashevsky. - K.: Vidavnicha group BHV, 2007 -352p.
4. Khusainov D.Ya. Introduced in modulyuvannya dynamical systems: Nav. posibnik / D.Ya. Khusainov, I. I. Kharchenko, A.V. Shatirko. K.: Kyiv National Taras Shevchenko University, 2010. 162p.
5. Aristov S.A. Simulation of economic systems: Proc. allowance. Ekaterinburg: Ural.gos.econ.un.-that. 2004. 123p.
6. Trub I.I. Object-oriented modeling in C++. Training course / I.Trub.- St. Petersburg: Peter, 2006. 411p.
7. Martin R., Martin M. Principles, patterns and techniques of flexible development in C# / R. Martin, M.Martin. - Symbol-Plus, 2011. – 768p.

**USE OF THE PROGRAM MODULE OF THE IMMATED MODELING OF THE WORK OF THE HOSPITAL CHAMBER IN THE STUDIES OF MASS-SERVICE SYSTEMS****Fedusenko Olena***Kyiv National Taras Shevchenko University*

**Abstract.** *The article is devoted to the problem of student study of mass service systems and simulation modeling, since these topics are quite complex and often do not have vivid examples. The article proposes a software module that will allow step by step simulation of mass service system with failures. In the program module developed by the author, the system of mass service is presented as a hospital chamber.*

*In the article the basic principles of building mass service systems and simulation modeling are considered. The main stages of simulation modeling are given. The task of simulation modeling of the hospital chamber, as a system of mass service with failures, is determined, the basic functional requirements for the software module are defined and the initial values of the values for modeling are given. The main functional requirements of the module is the possibility of step-by-step modeling and its display, both in the form of a table and in the form of a histogram, as well as the ability to read the initial data from the file and save the results to a text file or export them to Excel. In developing the software module, the author used the modern methodology of object-oriented programming SOLID according to which and built a diagram of the classes of the program module, which is given in the article. The use of this methodology allows, in the future, to expand and maintain the further functioning of the module rather easily, adding to it both new functionality and new modules.*

*The author describes the algorithm of the module operation in accordance with the class diagram and provides the results of the program module. The results of the program module work are presented both in tabular and graphical representation, in the form of a histogram. The obtained results, almost completely, coincide with the results presented in other studies, and the error is in the allowable range.*

*The developed software module can be used by students during the study of the discipline "Modeling systems", which will improve the quality of learning through the use of a visual example of simulation of mass service with failures.*

**Keywords:** *Mass service system, simulation modeling, object-oriented programming, SOLID.*