

Українська академія банківської справи
Національного банку України

Кафедра економічної кібернетики

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри

економічної кібернетики

канд. техн. наук, доцент

_____ С.М. Новак

«___» _____ 2007 р.

ДИПЛОМНА РОБОТА

на здобуття освітньо-кваліфікаційного рівня магістр
за спеціальністю 8.050102 «Економічна кібернетика»

Тема роботи: «Імітаційне моделювання ефективного обслуговування банкоматів та
їх оптимального розміщення»

Виконав студент 5 курсу, група МЕК-21

_____ О.О.Енська

«___» _____ 2007 р.

Керівник дипломної роботи

_____ С.О.Хайлук

«___» _____ 2007 р.

Тема роботи затверджена наказом по академії від «___» _____ 2007 р., №___

Суми - 2007



ЗМІСТ

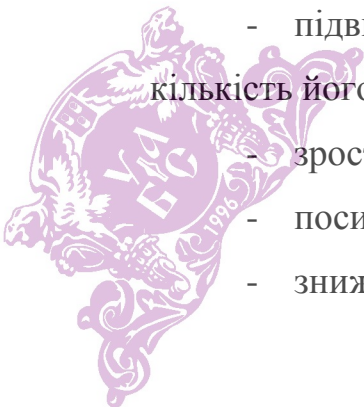
ВСТУП.....	5
1 НЕОБХІДНІСТЬ ТА ПЕРЕДУМОВИ МОДЕЛЮВАННЯ ЕФЕКТИВНОГО ОБСЛУГОВУВАННЯ БАНКОМАТІВ ТА ЇХ ОПТИМАЛЬНОГО РОЗМІЩЕННЯ.....	10
1.1 Сучасний стан використання банкоматів.....	10
1.2 Моделювання і його значення у вирішенні проблеми забезпеченні ефективної роботи банкоматів.....	19
1.3 Використання теорії масового обслуговування з метою розробки моделі ефективного обслуговування банкоматів.....	28
2 ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЕФЕКТИВНОГО ОБСЛУГОВУВАННЯ БАНКОМАТІВ ТА ЇХ ОПТИМАЛЬНОГО РОЗМІЩЕННЯ.....	38
2.1 Моделі визначення строків та обсягів забезпеченості банкоматів грошовими коштами.....	38
2.2 Визначення оптимального розміщення банкоматної мережі з використанням карти інтенсивності потоку клієнтів.....	61
2.3 Аналіз запропонованих моделей.....	65
3 РОЗРОБКА АВТОМАТИЗОВАНОГО ДОДАТКУ МОДЕЛІ ЕФЕКТИВНОГО ОБСЛУГОВУВАННЯ БАНКОМАТІВ ТА ЇХ ОПТИМАЛЬНОГО РОЗМІЩЕННЯ.....	67
3.1 Інформаційне, організаційне і програмне забезпечення.....	67
3.2 Проведення експериментування з моделлю на тестовому прикладі.....	71
3.3 Оцінка ефективності імітаційної моделі і аналіз результатів.....	80
ВИСНОВКИ.....	83
ПЕРЕЛІК ПОСИЛАНЬ.....	86
ДОДАТКИ.....	90



ВСТУП

Сучасна банківська система знаходиться у стані постійного і стрімкого розвитку. Збільшується кількість банків та спектр послуг, що надаються клієнтам. Зростає і кількість споживачів банківських послуг: якщо не так давно багато хто не довіряв банківським закладам і вважав більш надійним зберігати свої гроші вдома, то зараз все більше і більше людей змінюють своє ставлення до банків. Нині є цілком прийнятним і природним робити все для того, щоб гроші «працювали» і кожен одержував від цього свій зиск. Отже, у банків виникає необхідність збільшувати кількість філій і відділень, охоплюючи великі території: від одного міста до всієї України, а можливо і зарубіжжя. Така розгалужена банківська структура вимагає збільшення обсягів фінансування, що не завжди є можливим. Цю проблему можна вирішити шляхом систематизації та уніфікації банківських послуг, а також їх автоматизації, що скорочує витрати на утримання великої кількості кваліфікованого персоналу. У цьому питанні на допомогу банкам приходять новітні досягнення в сфері техніки та інформаційних технологій, які у тісному поєднанні дають той самий бажаний результат. Прикладом такого вдалого поєднання є використання банкоматів (АТМ, від англійського Automatic Teller Machine). І хоча на сьогодні застосування АТМ є дуже актуальним для обслуговування населення, українська система банкоматів ще дуже молода і знаходиться на етапі розвитку. Адже банкомати грають велику роль не тільки для окремих банків як засоби обслуговування великого потоку клієнтів, але й для банківської системи в цілому. Так, використання банкоматів має наступні переваги:

- підвищується конкурентоспроможність банку, а отже зростає кількість його клієнтів;
- зростає залучення грошових коштів у внутрішньобанківський обіг;
- посилюється керованість грошового обігу;
- знижується частка готівки в грошовій масі;



- скорочуються витрати на емісію грошей;
- підвищується роль банківської системи в розвитку економіки;
- з'являється додатковий стимул для переведення банківської системи на сучасні електронні технології;
- зростає попит на новітню комп'ютерну техніку, програмне забезпечення, системи зв'язку.

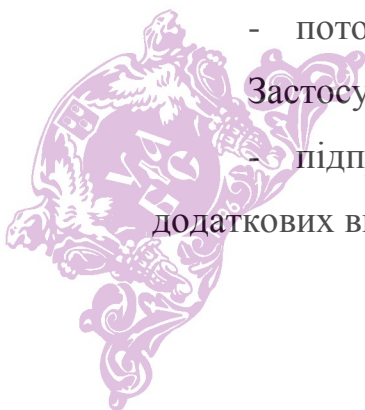
Технічний рівень банкоматів підвищується і спектр їх функцій постійно розширюється. На сьогоднішній день головним призначенням та застосуванням банкоматів у нашій країні є можливості клієнта банку дізнатися про стан свого рахунку, отримати готівку, роздрукувати чек. Однак, перелік операцій, які може здійснювати банкомат, значно ширший. При цьому, набір функцій, здійснюваних банкоматом, залежить також і від режиму його роботи.

За статистикою, найбільш поширеною послугою при використанні банкоматів в Україні на сьогодні є виплата заробітної плати, тобто видача грошей, оскільки здійснення цієї операції через касу нині вважається дорогим і незручним пережитком минулого. Більшість підприємств для оплати праці користуються послугами банку, знімаючи кошти з поточного рахунку. Для підприємств цей шлях є досить витратним і ризиковим [20]. У даному випадку підприємство несе витрати на:

- розрахунково-касове обслуговування (близько 1% фонду оплати праці);
- інкасацію (близько 2% фонду оплати праці);
- оплату праці касирів;
- депонування не отриманих співробітниками коштів;
- поточні платежі за роботу охоронної сигналізації тощо.

Застосування зарплатних проектів є вигідним для багатьох сторін:

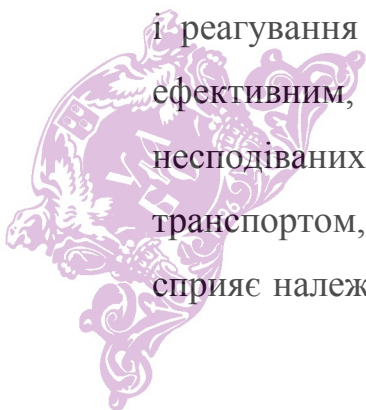
- підприємство отримує можливість позбутися усіх незручностей і додаткових витрат, пов'язаних з виплатою заробітної плати;



- співробітники можуть одержувати кошти у будь-який зручний для них час і у будь-якому місці, де є банкомат;
- банки отримують практично готову клієнтську базу, яку не потрібно заохочувати, а до того ж з якою можна працювати по різних програмах стосовно кредитування, депозитів і реклами різних банківських продуктів;
- для національної економіки впровадження зарплатних проектів має не менш важливе значення внаслідок того, що зменшується обсяг грошової маси в обігу, а до того ж висвітлюється значна частина реальних грошових доходів робітників, яка до цього видавалася в конвертах. Це в свою чергу сприяє розвитку податкової системи країни.

Зважаючи на все це, можна говорити, що застосування банкоматів є актуальним і перспективним напрямком банківської діяльності. Враховуючи те, що використання депозитних банкоматів та банкоматів з системами оперування готівкою поки що не є поширеним, частіше говорять про роботу банкоматів, що надають традиційні послуги з видачі грошей. Для таких банкоматів актуальною є проблема ефективної організації процесу заповнення їх готівкою.

Дана тема не достатньо висвітлена в літературних джерелах. В той же час, особливу увагу їй приділяли такі вчені, як Васин Н.С., Вересюк А., Зайцев О., Одарюк А. та інші. Варто зазначити, що дослідження даної проблеми, як правило проводиться самими банками, які потребують удосконалення процесу забезпечення банкоматів коштами внаслідок неефективної роботи мережі АТМ. Слід також зазначити, що найчастіше головним напрямком діяльності щодо налагодження функціонування мережі банкоматів є спостереження за роботою кожного банкомату в режимі он-лайн і реагування у відповідності до ситуації. Але такий шлях не є достатньо ефективним, адже він потребує додаткових витрат і залежить від багатьох несподіваних факторів, як то негаразди зі зв'язком чи проблеми з транспортом, погодними умовами тощо. Дія всіх цих факторів звичайно не сприяє належному обслуговуванню клієнтів банку. І це стосується не тільки



банкоматів, а й пластикових карток як головного супутнього продукту АТМ, а також депозитної, кредитної та іншої діяльності банку. Адже у кожній сфері банківського обслуговування клієнтів формується імідж всього закладу: іноді клієнт не в змозі об'єктивно оцінити позицію банку в цілому і робить висновок з конкретної ситуації. А тому для будь-якого банку є абсолютно необхідним мати ефективну, науково обгрунтовану і відпрацьовану на практиці методику, що дозволить правильно і найбільш зручно як для клієнтів, так і для банку організувати процес забезпечення банкоматів грошовими купюрами, а також допоможе розробити правильну структуру мережі банкоматів.

Тому у даній роботі буде розглядатися проблема ефективної організації банкоматної мережі та оптимізації процесу забезпечення банкоматів готівкою шляхом використання спеціальної методики статистичного моделювання функціонування банкоматних систем у поєднанні з методикою аналізу та прогнозування забезпеченості АТМ грошовими купюрами.

Метою роботи є розробка імітаційної моделі ефективного обслуговування банкоматів та їх оптимального розміщення.

Об'єктом дослідження виступають статистичні характеристики роботи банкоматів.

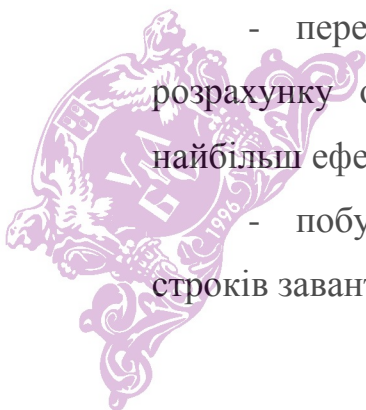
Предметом дослідження є методи і моделі ефективного обслуговування банкоматів та їх оптимального розміщення.

Основними завданнями даної роботи є:

- більш детальне вивчення вищезначеної проблеми і стану її практичного вирішення банківськими установами, зокрема, на прикладі роботи системи банкоматів «ПриватБанку» та ВАТ КБ «Надра»;

- перевірка на ефективність та адекватність різних методик розрахунку оптимальних сум і часу забезпечення банкоматів та вибір найбільш ефективної;

- побудова імітаційної моделі знаходження оптимальних сум та строків завантаження банкоматів готівкою;



- пошук методів організації ефективної структури банкоматної мережі та їх тестування;
- розробка рекомендацій по використанню моделі для оптимального розміщення банкоматів;
- розробка автоматизованого додатку запропонованої імітаційної моделі.

State Higher Educational Institution
"UKRAINIAN ACADEMY OF BANKING"
OF THE NATIONAL BANK OF UKRAINE

Державний вищий навчальний заклад
"УКРАЇНЬКА АКАДЕМІЯ БАНКІВСЬКОЇ СПРАВИ"
НАЦІОНАЛЬНОГО БАНКУ УКРАЇНИ



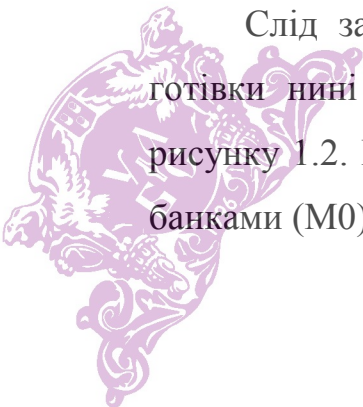
1 НЕОБХІДНІСТЬ ТА ПЕРЕДУМОВИ МОДЕЛЮВАННЯ ЕФЕКТИВНОГО ОБСЛУГОВУВАННЯ БАНКОМАТІВ ТА ЇХ ОПТИМАЛЬНОГО РОЗМІЩЕННЯ

1.1 Сучасний стан використання банкоматів

На сьогоднішній день економічна система країни знаходиться у стані розвитку. У цьому процесі важливу роль відіграє як реальний сектор, так і банківська система. Особливо важливим при цьому стає правильна організація емісії та обігу готівки в країні. Готівка, як правило, використовується при одержанні населенням своїх доходів та їх використанні (виплата заробітної плати, пенсій, стипендій, сплата за товари і послуги тощо). Вважається, що для ефективного функціонування економіки грошова маса в обігу повинна бути якомога меншою. Це досягається регулюванням емісії та безпосередньо обігу грошей. В Україні, згідно з континентальною моделлю [18], емісія грошей є прерогативою центрального банку. Що стосується побудови системи обігу готівки, варто зупинитися на досвіді Нідерландів [18], який можна вважати типовим для організації готівкового обігу країн євросони. Нідерландська модель обігу готівки представлена на рисунку 1.1.

Як можна побачити з даної схеми, у системах обігу готівки розвинених країн вагоме місце займають банкомати. Вони є головним, якщо не єдиним, інструментом передачі готівки фізичним особам, які, як зазначалося вище, є основними користувачами готівкових коштів. В той же час, банкомати є ефективним засобом зменшення обсягів грошової маси в обігу.

Слід зазначити, що Україна на шляху ефективної організації обігу готівки нині робить успіхи. Про це свідчить графік [2], зображений на рисунку 1.2. На цьому графіку показана динаміка питомої ваги грошей поза банками (M_0) в грошовій масі (M_1).



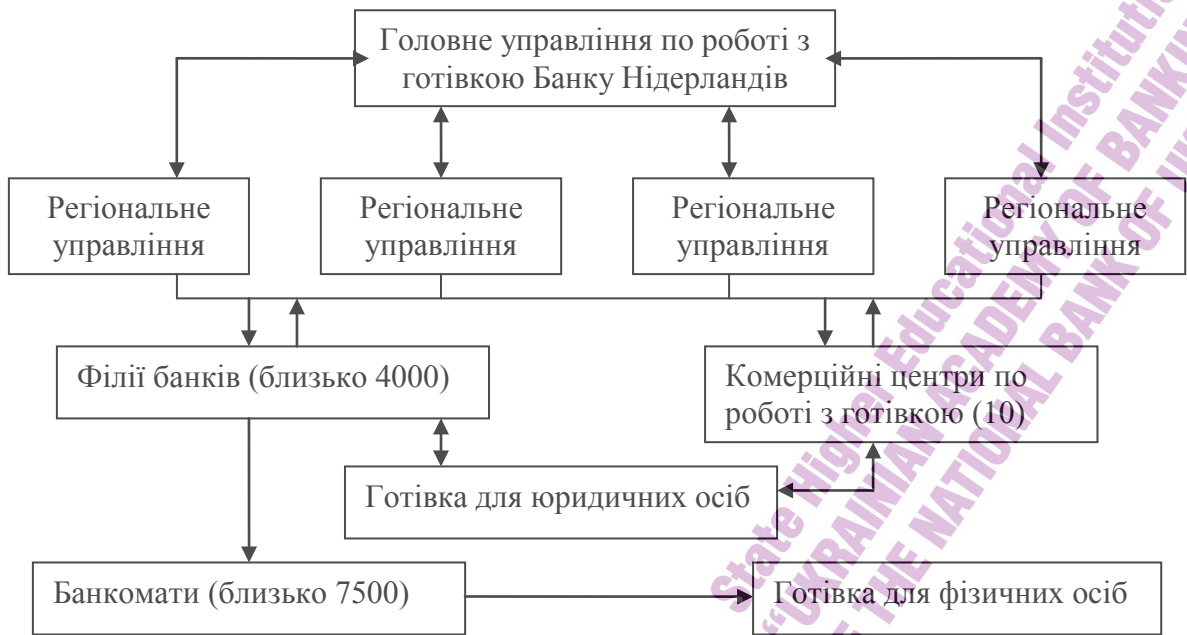


Рисунок 1.1 – Схема введення готівки в обіг Банком Нідерландів

Як видно з лінії тренда, зображеної на графіку, протягом декількох останніх років існує загальна тенденція до зниження вищезазначеного показнику.

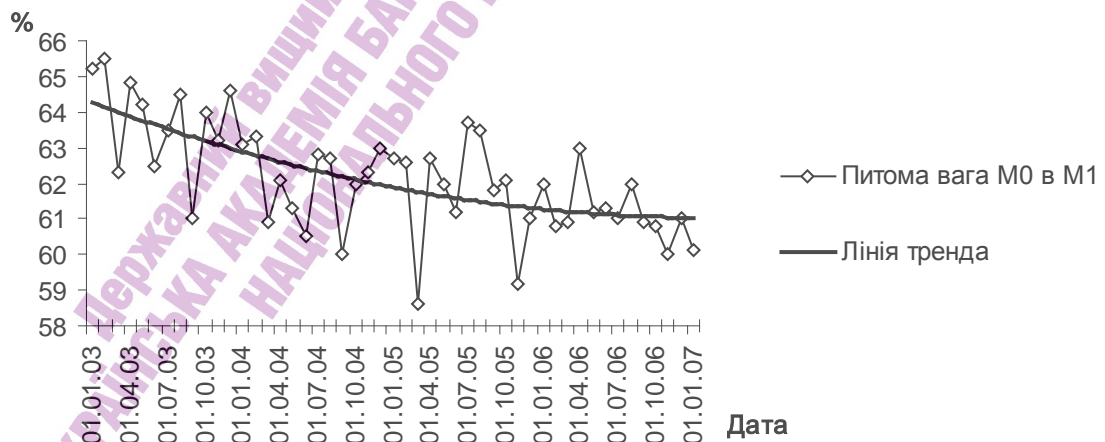


Рисунок 1.2 – Динаміка питомої ваги грошей поза банками (M0) в грошовій масі (M1)



Причинами такої ситуації є різні фактори, проте одним з основних є зростання кількості банкоматів і платіжних терміналів, встановлених банками. Про це свідчить і наступна статистика [35].

Так, кількість банкоматів, що обслуговують платіжні картки, за 2006 рік збільшилася на 3393 одиниці, або на 30% (з 11325 до 14718). Показники темпів приросту кількості банкоматів у 2006 році в розрізі платіжних систем наведено в таблиці 1.1.

Таблиця 1.1 – Темпи приросту кількості банкоматів у 2006 році в розрізі платіжних систем

Платіжні системи	Банкомати			
	на 01.01.2006	на 01.01.2007	Приріст	
			одиниць	%
НСМЕП	367	557	190	52
Укркарт	778	1049	271	35
Одноемітентні	1837	2236	399	22
MasterCard	9879	3484	3605	36
VISA	9289	12236	2947	32
Усього	11325	14718	3393	30

Число банкоматів у розрахунку на мільйон чоловік населення України протягом року збільшилося на 75 одиниць і на 01.01.2007 р. становило 316 одиниць (на кінець 2005 року – 241). Цей показник в Україні утричі нижчий, ніж у середньому в країнах Європейського Союзу (за даними Європейського центрального банку на кінець 2005 року – 735 одиниць). Водночас слід зазначити, що вже на кінець 2005 року в Україні на мільйон чоловік населення було встановлено більше банкоматів, ніж в окремих нових країнах Європейського Союзу, зокрема Румунії (201 одиниця) та Польщі (230 одиниць). Якщо така динаміка розвитку банкоматних мереж, яка спостерігалася в Україні протягом останніх 3 – 4 років, збережеться і надалі, то вже за 5 років можна очікувати досягнення Україною нинішніх

середньоєвропейських показників за кількістю встановлених банкоматів на мільйон чоловік населення.

Якщо порівнювати кількість банкоматів із кількістю емітованих платіжних карток, то за станом на 01.01.2007 р. 14718 встановлених в Україні банкоматів обслуговували 32,5 млн. емітованих платіжних карток, тобто в середньому на один банкомат припадало 2206 карток, що на 18 платіжних карток більше, ніж у 2005 році (2188шт.).

На кінець 2006 року 99 банків мали власні банкомати (у 2005 році — 90 банків). З них 18 встановили понад 200 банкоматів. У таблиці 1.2 наведено українські банки-лідери за кількістю встановлених банкоматів станом на 01.01.2007 року.

Таблиця 1.2 – Банки України – лідери за кількістю встановлених банкоматів станом на 01.01.2007 року

№ п/п	Назва банку	01.01.2006 р.	01.01.2007 р.	Приріст за 2006 р.	
				одиниць	%
1	Приватбанк	2934	3429	495	16,9
2	Райффайзен Банк Аваль	1 411	1 819	408	28,9
3	Промінвестбанк	1 175	1380	205	17,4
4	«Надра»	593	735	142	23,9
5	Укрсоцбанк	500	675	175	35,0
6	Укрсиббанк	427	612	185	43,3
7	Ощадбанк	509	565	56	11,0
8	Укргазбанк	367	480	113	30,8
9	ПУМБ	291	472	181	62,2
10	Укрексімбанк	308	411	103	33,4
11	Укрпромбанк	201	295	94	46,8
12	Експрес-банк	225	274	49	21,8
13	«Фінанси та кредит»	183	257	74	40,4

Продовження таблиці 1.2

№ п/п	Назва банку	01.01.2006 р.	01.01.2007 р.	Приріст за 2006 р.	
				одиниць	%
14	«Форум»	104	224	120	115,4
15	ВАБанк	129	222	93	72,1
16	Правекс-банк	215	219	4	1,9
17	«Хрещатик»	106	217	111	104,7
18	Кредитпромбанк	126	202	76	60,3

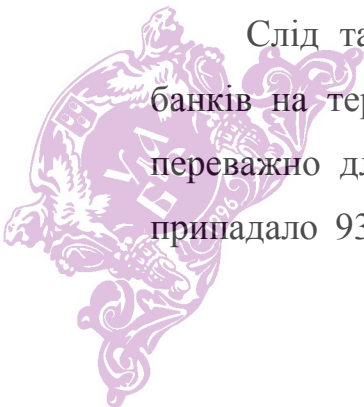
Разом на ці банки припадало майже 85% усіх банкоматів (12488 од.), що функціонували в Україні у 2006 році.

За чисельністю встановлених банкоматів лідерами серед банків залишаються Приватбанк (3429 од.), Райффайзен Банк Аваль (1819 од.) та Промінвестбанк (1380 од.), які разом встановили понад 45% усіх банкоматів в Україні. Темпи приросту кількості банкоматів у цих банків 2006 року становили відповідно 16,9% (495 од.), 28,9% (408 од.) і 17,4% (205 од.).

Динамічніше за всіх (із банків, що встановили понад 200 банкоматів) 2006 році розвивали свою банкоматну мережу банки «Форум» і «Хрещатик», які більш як подвоїли кількість встановлених ними банкоматів. П'ять банків за станом на 01.01.2007 р. встановили понад 100 банкоматів. Це, зокрема, Кредобанк (161 од.), Брокбізнесбанк (149 од.), Імекс-банк (139 од.), Індустріалбанк (127 од.) і банк «Південний» (120 од.).

Більш наочно динаміку зростання кількості встановлених банкоматів представлено на рисунку 1.3.

Слід також зазначити, що держателі платіжних карток українських банків на території України протягом минулого року використовували їх переважно для зняття готівки. Так, у 2006 році на ці операції в Україні припадало 93,6% (426,5 млн. шт.) від загальної кількості внутрідержавних



операцій, що становило 97,8% (145053,3 млн.) від загального обороту в Україні за платіжними картками, емітованими українськими банками.

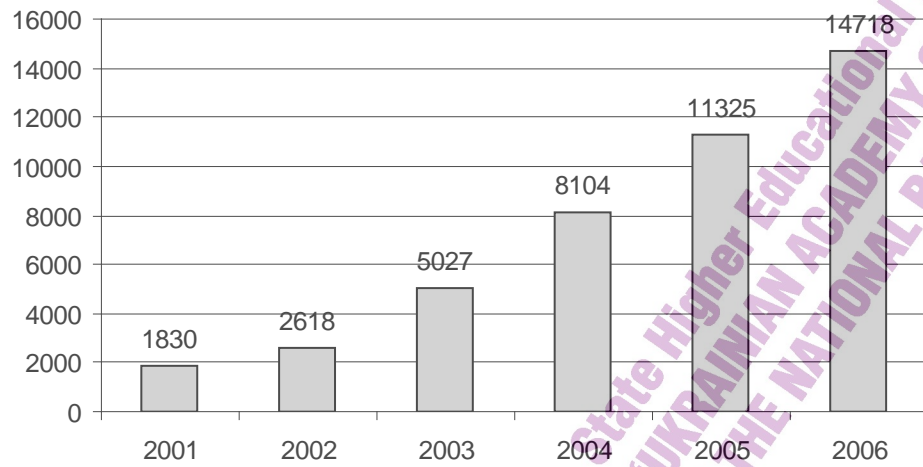


Рисунок 1.3 – Динаміка зростання кількості встановлених банкоматів в Україні (одиниць)

Дані про кількість і суми операцій із використанням платіжних карток у 2006 році наведено у таблиці 1.3.

Таблиця 1.3 – Дані про кількість і суми операцій із використанням платіжних карток у 2006 році

№ п/п	Використання платіжних карток	Оплата товарів (послуг)		Зняття готівки		Усього	
		Кількість операцій, млн. шт.	Сума операцій, млн. грн.	Кількість операцій, млн. шт.	Сума операцій, млн. грн.	Кількість операцій, млн. шт.	Сума операцій, млн. грн.
1	Емітованих українськими банками в Україні	29,4	3289,4	426,5	145053,3	455,9	148342,7
2	Емітованих банками-нерезидентами в Україні	1,5	1101,5	4,3	3549,0	5,8	4650,5

Продовження таблиці 1.3

№ п/п	Використання платіжних карток	Оплата товарів (послуг)		Зняття готівки		Усього	
		Кількість операцій, млн. шт.	Сума операцій, млн. грн.	Кількість операцій, млн. шт.	Сума операцій, млн. грн.	Кількість операцій, млн. шт.	Сума операцій, млн. грн.
3	Емітованих українськими банками за кордоном	1,6	1759,5	1,1	2543,6	2,7	4303,1
4	Усього в Україні	30,9	4390,9	430,8	148602,3	461,7	152993,2
5	Усього за картками, емітованими українськими банками	31,0	5048,9	427,6	147596,9	458,6	152645,8

Співвідношення способів використання платіжних карток в Україні відображено на рисунку 1.4.



Рисунок 1.4 – Використання платіжних карток, емітованих українськими банками, в Україні



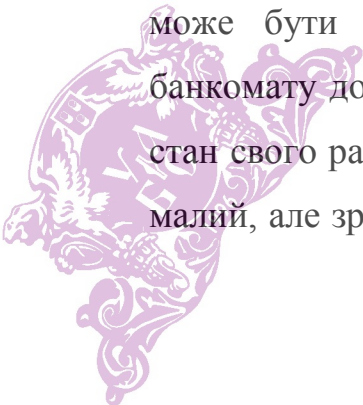
Сучасні мережі банкоматів є досить розгалуженими, що ускладнює їх обслуговування і контроль. Окрім того, банкомати можуть працювати у різних режимах, від чого також залежать витрати на них. З цієї точки зору банкомати поділяються на дві групи:

- оф-лайн (off-line);
- он-лайн (on-line).

Банкомати, що працюють в режимі оф-лайн – це найбільш поширена група банкоматів, оскільки вони працюють з клієнтами автономно, без постійного зв'язку з розрахунковим центром (свого банку чи системи карток). Це повноцінний банкомат. Основна проблема при обслуговуванні оф-лайнових систем – це занесення в їх пам'ять (найчастіше на магнітний диск) бази про анульовані (загублені, викрадені) картки або такі, термін дії яких уже вичерпався.

Собівартість оф-лайнових банкоматів відносно низька, бо їх можна обслуговувати «вручну»: один-два рази на добу оператор банкоматів виймає протокол проведених операцій (це, як правило, паперова стрічка дублікатів квитанцій та магнітний диск чи флеш-пам'ять з відповідними записами) для передачі до розрахункового центру відповідного банку (мережі банків чи системи карток). До пам'яті заноситься база карток, за якими операції проводити не можна. Проте ризик у зловживанні пластиковими картками для таких автоматів, зрозуміло, вищий.

Робота в режимі он-лайн передбачає постійний зв'язок через телекомунікаційний канал банкомату із розрахунковим центром «свого» банку (мережі банків) чи процесинговим центром країни (системи карток). Відповідно кожна операція одразу стає відома у розрахунковому центрі і може бути перевірена та підтверджена. Постійний «гарячий» зв'язок банкомату дозволяє власнику картки в будь-який момент брати довідки про стан свого рахунка. Ризик у зловживанні картками для таких автоматів дуже малий, але зростають накладні витрати на організацію телекомунікацій. Для



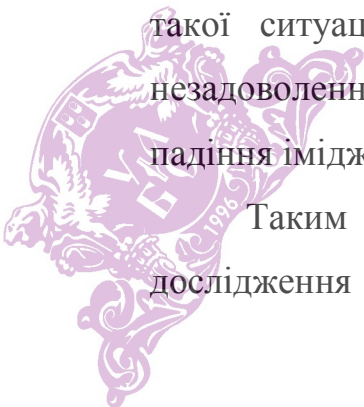
країн із поганим чи дорогим рівнем телекомунікацій он-лайновий зв'язок з банкоматами по всій країні не використовується.

Однак, у будь-якому випадку, незалежно від режиму зв'язку, головною функцією банкомату є видача грошей.

Враховуючи ці умови, кожен банк постійно мусить збільшувати витрати коштів і часу, а також залучати додаткові кадри. Але, звичайно, метою будь-якої комерційної установи є зменшення витрат і підвищення прибутку. Для досягнення цієї мети можна використовувати два способи: екстенсивний і інтенсивний. Перший спосіб означає збільшення обсягів залучення ресурсів. Та цей спосіб не завжди є доступним у повній мірі кожному банку та зазвичай є не самим ефективним. Більший інтерес викликає інтенсивний спосіб, коли обсяг використовуваних ресурсів майже не збільшується, а підвищується раціональність підходів до використання наявних ресурсів. Виходячи з того, що у будь-якому випадку банку доводиться збільшувати обсяги фінансування обслуговування банкоматів, то слід більшу увагу приділити такому питанню, як визначення оптимальних сум коштів, які завантажуються до банкоматів.

Кожен банк використовує для цього різні методики, які можуть бути розроблені як самими банками, так і сторонніми дослідниками. Та здебільшого банки користуються таким способом контролю забезпеченості банкоматів готівкою, як моніторинг мережі банкоматів в режимі он-лайн і надання до неї грошей за потребою. Такий спосіб не має серйозних недоліків, якщо є визначеним ліміт коштів для забезпечення певної кількості банкоматів цілих структурних підрозділів банку, та він не оперує інформацією про ліміти для кожного банкомату окремо. Це призводить до такої ситуації, коли робота банкоматів не є ефективною і викликає незадоволення користувачів пластикових карток. Це, в свою чергу, веде до падіння іміджу банку і втрат можливого прибутку.

Таким чином, вище описана проблема потребує ретельного дослідження і пошуку спеціальних методик вирішення, які мають бути



обґрунтовані науково і випробувані на практиці. Подібні методика дадуть можливість визначати суми коштів, які повністю забезпечать потік клієнтів, а до того ж скоротити додаткові витрати як на забезпечення банкоматів грошовими купюрами, так і на їх обслуговування.

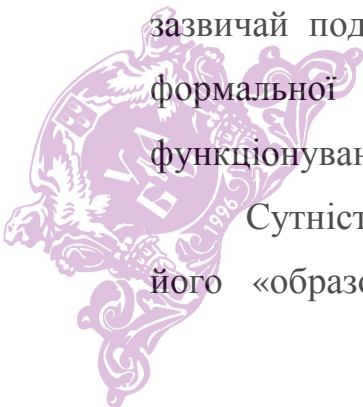
1.2 Моделювання і його значення у вирішенні проблеми забезпеченні ефективної роботи банкоматів

Економіка є складною системою, яка містить безліч об'єктів, суб'єктів, процесів і факторів, що на них впливають. Існує ряд економічних законів, які пояснюють функціонування і взаємодію елементів економічної системи, але навіть вони іноді не в змозі точно спрогнозувати ту чи іншу ситуацію і її наслідки. Економічна теорія пропонує перелік методів, як загальнонаукових, так і специфічних, що покликані допомогти вирішити деякі проблеми, які виникають, або передбачити і запобігти їх виникненню. Серед них найбільш популярними нині є методи математичного та статистичного моделювання, які у поєднанні дають найкращі результати.

Термін «модель» походить від латинського слова «modulus», що в перекладі означає «зразок», «норма», «міра». У загальному розумінні моделлю є об'єкт, що заміщує оригінал і відбиває його найважливіші риси й властивості в рамках певного дослідження, для досягнення поставленої мети та за обраної системи гіпотез [8].

Математична модель – це абстракція реальної дійсності, в якій відношення між реальними елементами, а саме ті, що цікавлять дослідника, замінені відношеннями між математичними категоріями. Ці відношення зазвичай подаються у формі рівнянь та (або) нерівностей, відношеннями формальної логіки між показниками (змінними), які характеризують функціонування реальної системи, що моделюється [8].

Сутність методології моделювання полягає в заміні вихідного об'єкта його «образом» – математичною моделлю – і подальшим вивченням

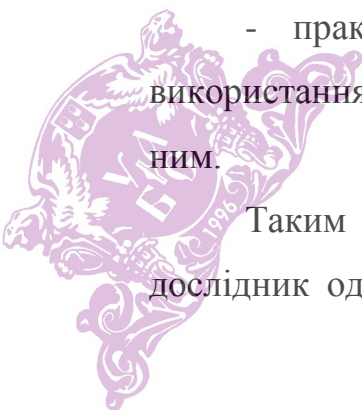


(дослідженням) моделі на підставі аналітичних методів та обчислювально-логічних алгоритмів, які можуть реалізовуватися за допомогою комп'ютерних програм. Безперечною перевагою моделювання є те, що робота проводиться не із самим об'єктом (явищем, процесом), а з його моделлю, і це дає можливість відносно швидко і безпечно досліджувати його основні (суттєві) властивості та поведінку у будь-яких можливих ситуаціях. Водночас обчислювальні експерименти з моделями об'єктів дозволяють, спираючись на потужність сучасного математичного і технічного інструментарію, ретельно і та досить глибоко вивчати об'єкт у достатньо детальному вигляді, що недоступно суто теоретичним підходам.

У моделюванні використовуються методи абстракції, аналогії, гіпотез, індукції, дедукції, аналізу, синтезу та інші. Саме використання цих методів забезпечує необхідну глибину дослідження реального об'єкту з багатьох сторін, у різних ситуаціях. Процес моделювання включає наступні етапи:

- вивчення властивостей реального об'єкта дослідження і створення моделі з наданням їй тих характеристик оригіналу, які мають сенс у даному дослідженні, і відмовленням від усіх інших властивостей, що у даній ситуації не мають суттєвого значення;
- проведення модельних експериментів, в яких модель постає як самостійний об'єкт і реагує на свідомі зміни умов її функціонування; систематизація даних про поведінку моделі в різних ситуаціях;
- перенесення отриманих на попередньому етапі знань з моделі на оригінал за певними правилами; коригування відомостей про об'єкт з урахуванням тих властивостей оригіналу, що не знайшли відображення під час моделювання або були деформовані;
- практична перевірка одержаних за допомогою моделей знань та використання їх для побудови узагальнюючої теорії об'єкта чи управління ним.

Таким чином, використовуючи економіко-математичне моделювання, дослідник одержує можливість оцінити та спрогнозувати певні економічні



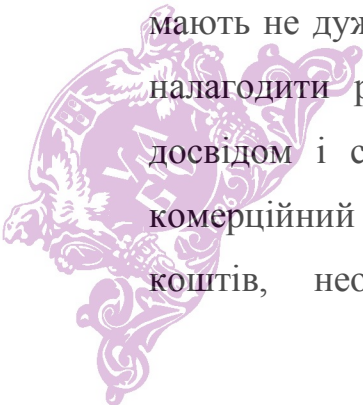
явища, а змінюючи параметри моделі та умови її функціонування – знайти найбільш прийнятний варіант, що відповідає отриманню оптимальних результатів на практиці. Це є дуже зручним, враховуючи специфіку економіки як такої, що виключає можливість проведення експериментів, які є найкращим способом дослідження і дозволяють отримати найбільш реальні дані. Моделювання, по суті, є тим самим експериментом, але воно не створює несприятливих наслідків і в той самий час дозволяє отримати результати, найбільш наближені до реальних.

Для моделювання економічних процесів можуть використовуватися різноманітні підходи та математичні теорії. Це залежить від досліджуваної ситуації та суб'єктивного вибору особи, що проводить моделювання, а також від того, які дані використовуються у якості вхідних параметрів і які результати ми очікуємо одержати.

Для вирішення задачі ефективного забезпечення банкоматів готівкою на практиці різними банками використовуються різні моделі й методики. Проте, їх об'єднують такі загальні принципи:

- гроші не повинні перебувати в банкоматі надто довго, адже вони можуть бути використані іншим чином в роботі банку і принести додатковий прибуток;
- банк не має дуже часто завантажувати кошти в банкомат, тому що це призводить до зайвих витрат на транспорт і оплату праці персоналу, який обслуговує банкоматами.

Дуже часто українські банки не використовують жодних методик чи моделей для розрахунку сум коштів, які необхідні для забезпечення функціонування їх мереж банкоматів. До таких банків належать такі, що мають не дуже великі банкоматні мережі, оскільки їх співробітники можуть налагодити роботу банкоматів, користуючись при цьому лише власним досвідом і статистикою попереднього використання готівки. Наприклад, комерційний банк «Надра» не має спеціальної методики для розрахунку сум коштів, необхідних для завантаження банкоматів. Обслуговування

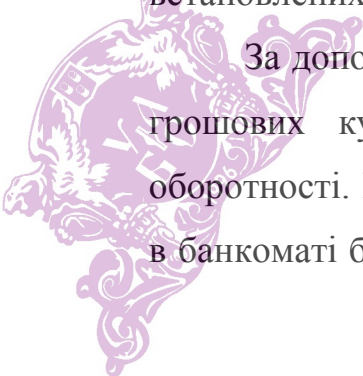


банкоматів входить до обов'язків окремих співробітників, які мають усі відомості про користування банкоматами свого банку за попередні періоди і забезпечують безперебійну роботу мережі переважно шляхом моніторингу. При завантаженні готівкою банкоматів вони користуються набутим досвідом та інтуїцією. Слід зазначити, що для мережі такого масштабу, як у банку «Надра», такий спосіб забезпечення банкоматів готівкою виявляється досить ефективним і пошук або розробка інших рішень даної задачі є недоцільним, оскільки витрати на це можуть перевищити економію на вивільнених коштах. Проте існують банки, банкоматні мережі яких значно ширші, ніж у банку «Надра», мають дуже велику кількість банкоматів та охоплюють значні площі. подібні банки мусять використовувати спеціальні методи для розрахунку коштів, необхідних для забезпечення мереж банкоматів, оскільки без таких методів банк буде мати надмірні витрати на обслуговування банкоматів та забезпечення їх коштами. А у великих масштабах такі витрати будуть значно впливати на роботу всього банку. Тому банки з великими мережами банкоматів використовують спеціальні методики та моделі, які найчастіше розробляють самі.

Прикладом моделі, використовуваної на практиці для вирішення вищеописаної задачі ефективного забезпечення банкоматів готівкою може слугувати модель, яка застосовується «ПриватБанком». Даний банк є безперечним лідером серед українських банків за кількістю банкоматів, і у 2007 році у мережі обслуговування пластикових карт цього банку нараховується понад 3,5 тис. банкоматів.

Пропонована банком методика дозволяє визначити ліміти залишків грошових коштів у банкоматах та здійснювати контроль виконання встановлених лімітів.

За допомогою моделі «ПриватБанку» встановлюються ліміти залишків грошових купюр у банкоматах за нормативних значень коефіцієнту оборотності. Він встановлюється таким чином, щоб час перебування грошей в банкоматі був оптимальний, тобто не надто довгий, щоб гроші працювали,



але й не надто короткий. щоб не підвищувались витрати на обслуговування банкоматів.

Окрім коефіцієнту оборотності, в моделі враховується величина виданих засобів за попередній період. Ця величина може змінюватися з кожним наступним періодом, в залежності від динаміки використання банкоматів клієнтами банку. Якщо банкомат активно працює на видачу наявних засобів, у нього буде значний кредитовий оборот і відповідно для нього буде збільшено ліміт.

Інтегральний показник визначення ліміту залишку засобів в банкоматах виглядатиме таким чином:

$$Lim = \frac{K_{об.норм.} * Kред.}{T} \quad (1.1)$$

де Lim – ліміт залишку засобів в банкоматах (гр.од.);

$K_{об.норм.}$ – нормативний коефіцієнт оборотності (днів);

$Kред.$ – сумарний кредитовий оборот (оборот, що викликає зменшення залишку в банкоматах) (гр.од.)

T – кількість календарних днів, за які розрахований кредитовий оборот (днів).

При розрахунку ліміту залишку засобів в банкоматах проводиться щомісячний аналіз коефіцієнтів оборотності. Нормативний коефіцієнт оборотності встановлюються рівним найкращому значенню даного коефіцієнта, окремо для кожного підрозділу банку.

Оскільки немає можливості обліку величини обов'язкового загального незнижуваного залишку в кожному банкоматі, обчислюється поправка, яка додається до розрахункової величини ліміту. Розрахунок поправки здійснюється таким чином: за три місяці обчислюється сальдо (1004А-2920П) за кожен робочий день в цілому за кожним філіалом, на балансі якого є



банкомати. Мінімальне ненульове позитивне значення додається до розрахункової величини ліміту.

Розрахункова величина ліміту і величина поправки округляється до цілих тисяч – у більший або менший бік, залежно від величини розрахункового значення.

Для розрахунку обсягу операцій по банкоматах, а також при розрахунку коефіцієнта оборотності, використовуються залишки і обороти по балансових рахунках, наведених в таблиці 1.4.

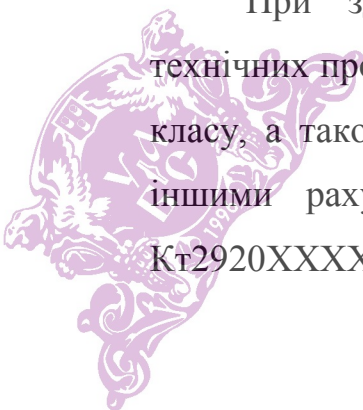
Таблиця 1.4 – Рахунки, які використовуються для обчислення обсягу операцій по банкоматах та розрахунку коефіцієнта оборотності

Балансові рахунки	Найменування рахунку згідно плану рахунків
1004	Банкноти і монети в банкоматах
2920	Транзитний рахунок по операціях, здійснених платіжними картами через банкомат
2924	Транзитний рахунок по операціях з платіжними картами

Для кожного банкомату відкриті індивідуальні рахунки 1004 і 2920. Визначення величини ліміту проводиться по кожному банкомату окремо і підсумковий ліміт по регіональному підрозділу визначається підсумовуванням лімітів за всіма банкоматами.

Рахунки ідентифікуються за допомогою спеціального файлу (таблиці бази даних). Зв'язок рахунків 1004 і 2920 в межах одного філіалу здійснюється по умовному номеру банкомату.

При здійсненні розрахунків кредитові обороти не враховують технічних проводок, за яких рахунок 1004 кореспондує з рахунками першого класу, а також проводки, при яких рахунки 1004 і 2920 кореспондують з іншими рахунками 1004 і 2920 (наприклад, Дт2920XXXXXXXX100 / Кт2920XXXXXXXX200).



Для коректного відображення фактичного залишку засобів в банкоматах при контролі виконання лімітів активний залишок по рахунку 1004 зменшується на величину пасиву по рахунках 2920.

Коефіцієнт оборотності (*Коб.*) розраховується за формулами (1.2) і (1.3).

Для регіональних підрозділів:

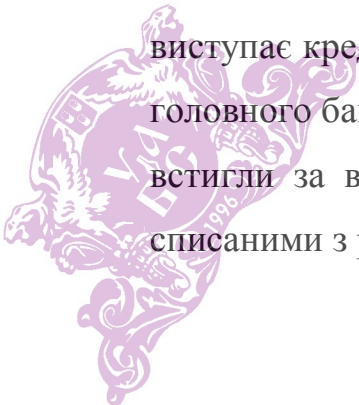
$$K_{об} = \frac{\text{Сумарне сальдо за інтервал}}{\text{Сумарний кредитовий оборот за інтервал}} = \frac{\sum (1004A - 2920П)}{\sum (Дт2924 / Кт2920)} \quad (1.2)$$

Для головного банку:

$$K_{об} = \frac{\text{Сумарне сальдо за інтервал}}{\text{Сумарний кредитовий оборот за інтервал}} = \frac{\sum (1004A - 2920П)}{\sum (Дт2924 / Кт1004)} \quad (1.3)$$

Коефіцієнт оборотності – коефіцієнт, що враховує регіональні особливості кожного регіонального підрозділу або головного банку. Даний коефіцієнт відрізняється для різних регіонів та залежить від ступеню їх ділової активності.

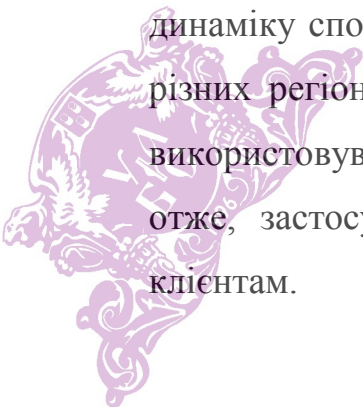
Чим вище абсолютне значення коефіцієнту оборотності, тим нижче інтенсивність обігу засобів в банкоматі. У даному розрахунку як показник, що визначає ліміт залишку засобів в банкоматах в грошовому вираженні, виступає кредитовий оборот по рахунках 2920 для структур банку і 1004 для головного банку. Він характеризує ту величину грошових коштів, які реально встигли за вказаний період поступити в банкомат (надходження) і бути списаними з рахунків банкомату (витрати).



Нормативний коефіцієнт оборотності розраховується Фінансовим Департаментом (управлінням лімітами), виходячи з фактичної оборотності засобів за період не менше трьох місяців. Він приймається рівним найкращому (найменшому) коефіцієнту з трьох.

Ліміт залишку засобів в банкоматах, розрахований за методикою, описаною вище, встановлюється на регіональні підрозділи, самостійні філіали і головний банк системи «ПриватБанку» щомісячно. Розрахований ліміт надається Фінансовим Департаментом на узгодження головному банку і затверджується спеціальним розпорядженням. Зміна ліміту можлива за ініціативою регіональних підрозділів і філіалів, Фінансового Департаменту. У цьому випадку структури банку направляють службові записки на узгодження, після чого, управління лімітами вносить зміни в ліміти структур банку. Управління лімітами банку щодня здійснює розрахунок виконання лімітів структурами банку і надає отриману інформацію керівництву банку. Залежно від ситуації на міжбанківському ринку фактичне значення залишку засобів в банкоматах для контролю виконання лімітів може розраховуватися за двома способами: як середнє сальдо за рахунком 1004 (з урахуванням рахунку 2920) за певний період або як фактичне сальдо за тими ж рахунками за один минулий день (приклад форми наведено у додатку А). Інформація про те, який з механізмів контролю застосовуватиметься і починаючи з якого періоду, доводиться до структур банку розпорядженнями.

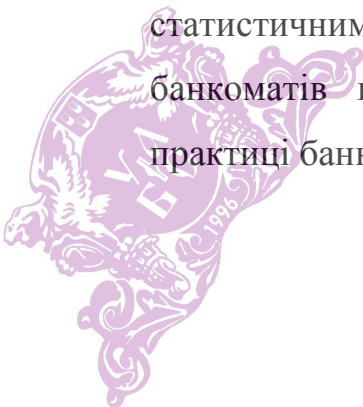
Таким чином, за даною моделлю розрахунок сум грошей, якими необхідно забезпечувати банкомати для ефективного обслуговування клієнтів банку, здійснюється централізовано і надається структурним підрозділам банку у вигляді нормативів. Така методика дозволяє врахувати динаміку споживання коштів, залежно від місця розташування банкоматів у різних регіонах, а також дає можливість якомога ефективнішим способом використовувати кошти для одержання максимально можливого прибутку. А отже, застосування подібної моделі дає переваги як банку, так і його клієнтам.



Однак, дана модель не враховує невизначеності процесу використання банкомату як процесу масового обслуговування, а отже необхідним є пошук інших моделей і методів, які б могли бути більш ефективними у цьому випадку. До того ж, моделі, використовувані банками на практиці, найчастіше є аналітичними, і не завжди мають розв'язки.

Слід зазначити, що досить ефективним для розв'язання подібних задач є імітаційне моделювання. Імітаційне моделювання – це метод, що дозволяє будувати моделі, процеси, що описують, так, як вони проходили б насправді. Таку модель можна «програти» в часі як для одного випробування, так і заданої їх множини. При цьому результати визначатимуться випадковим характером процесів. За цими даними можна отримати достатньо стійку статистику. При цьому система, що вивчається, замінюється імітатором і з ним проводяться експерименти з метою отримання інформації про цю систему. Експериментування з імітатором називають імітацією. Імітаційне моделювання є окремим випадком математичного моделювання. Існує клас об'єктів, для яких з різних причин не розроблені аналітичні моделі, або не розроблені методи рішення отриманої моделі, або аналітичні моделі не дають достатньо адекватних результатів. В цьому випадку математична модель замінюється імітатором або імітаційною моделлю. Імітаційна модель – це логіко-математичний опис об'єкту, який може бути використаний для експериментування на комп'ютері в цілях проектування, аналізу і оцінки функціонування об'єкту. [42]

У даній роботі детально буде розглянуто модель, що використовує теоретико-ймовірнісний підхід і описує процес роботи і обслуговування банкоматів з позиції теорії масового обслуговування у поєднанні зі статистичними методами для прогнозування часу та сум завантаження банкоматів грошовими коштами, та модель, яка використовується на практиці банком для вирішення аналогічної задачі.



1.3 Використання теорії масового обслуговування з метою розробки моделі ефективного обслуговування банкоматів

Розглянемо процес обслуговування клієнтських заявок банкоматом з позиції теорії масового обслуговування. Наведемо деякі поняття [9].

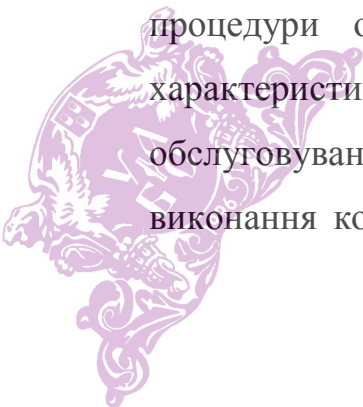
Системи масового обслуговування – це такі системи, в які у випадкові моменти часу поступають заявки на обслуговування, заявки, що при цьому поступили, обслуговуються за допомогою каналів обслуговування, що є у розпорядженні системи.

У теорії систем масового обслуговування обслуговуваний об'єкт називають вимогою. У загальному випадку під вимогою звичайно розуміють запит на задоволення деякої потреби.

Дисципліна черги – це важливий компонент системи масового обслуговування, він визначає принцип, відповідно до якого вимоги, що поступають на вхід обслуговуючої системи, підключаються з черги до процедури обслуговування. Найчастіше використовуються дисципліни черги, визначувані наступними правилами:

- першим прийшов – перший обслуговується;
- прийшов останнім – обслуговується першим;
- випадковий відбір заявок;
- відбір заявок по критерію пріоритетності;
- обмеження часу очікування моменту настання обслуговування (має місце черга з обмеженим часом очікування обслуговування, що асоціюється з поняттям «допустима довжина черги»).

Механізм обслуговування визначається характеристиками самої процедури обслуговування і структурою обслуговуючої системи. До характеристик процедури обслуговування відносяться: тривалість процедури обслуговування і кількість вимог, що задовольняються в результаті виконання кожної такої процедури. Для аналітичного опису характеристик



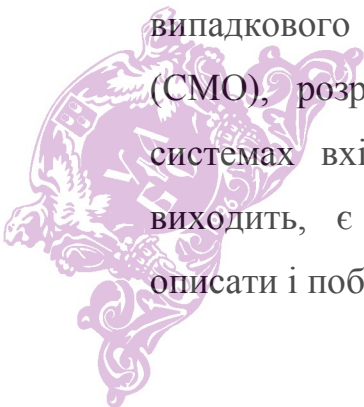
процедури обслуговування оперують поняттям розподіл «вірогідності часу обслуговування вимог».

Слід зазначити, що час обслуговування заявки залежить від характеру самої заявки або вимог клієнта і від стану і можливостей обслуговуючої системи. Структура обслуговуючої системи визначається кількістю і взаємним розташуванням каналів обслуговування. Перш за все слід підкреслити, що система обслуговування може мати не один канал обслуговування, а декілька; система такого роду здатна обслуговувати одночасно декілька вимог. В цьому випадку всі канали обслуговування пропонують одні і ті ж послуги, і, отже, можна стверджувати, що має місце паралельне обслуговування.

Система обслуговування може складатися з декількох різнотипних каналів обслуговування, через які повинне пройти кожна обслуговувана вимога, тобто в обслуговуючій системі процедури обслуговування вимог реалізуються послідовно. Механізм обслуговування визначає характеристики потоку вимог, що виходить.

Предметом теорії масового обслуговування є встановлення залежності між чинниками, що визначають функціональні можливості системи масового обслуговування, і ефективністю її функціонування. В більшості випадків всі параметри, що описують системи масового обслуговування, є випадковими величинами або функціями, тому ці системи відносяться до стохастичних систем.

Випадковий характер потоку заявок (вимог), а також, в загальному випадку, і тривалості обслуговування призводить до того, що в системі масового обслуговування відбувається випадковий процес. По характеру випадкового процесу, що відбувається в системі масового обслуговування (СМО), розрізняють системи марківські і немарківські. У марківських системах вхідний потік вимог і потік обслужених вимог (заявок), що виходить, є пуассоновськими. Пуассоновські потоки дозволяють легко описати і побудувати математичну модель системи масового обслуговування.



Дані моделі мають достатньо прості рішення, тому більшість відомих додатків теорії масового обслуговування використовує марківську схему. У разі немарківських процесів задачі дослідження систем масового обслуговування значно ускладнюються і вимагають застосування статистичного моделювання, чисельних методів з використанням ЕОМ.

Незалежно від характеру процесу, що протікає в системі масового обслуговування, розрізняють два основні види СМО:

- системи з відмовами, в яких заявка, що поступила в систему в мить, коли всі канали зайняті, дістає відмову і зразу ж покидає чергу;
- системи з очікуванням (чергою), в яких заявка, що поступила в мить, 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.

У системах з обмеженим очікуванням може обмежуватися:

- довжина черги;
- час перебування в черзі.

У системах з необмеженим очікуванням заявка, що стоїть в черзі, чекає обслуговування необмежено довго, тобто поки не підійде черга.

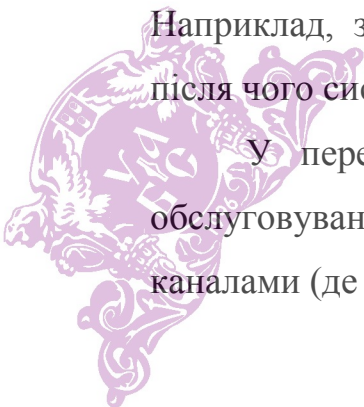
Всі системи масового обслуговування розрізняють по числу каналів обслуговування:

- одноканальні системи;
- багатоканальні системи.

Приведена класифікація СМО є умовною. На практиці найчастіше системи масового обслуговування виступають як змішані системи.

Наприклад, заявки чекають початку обслуговування до певного моменту, після чого система починає працювати як система з відмовами.

У переважній більшості випадків на практиці системи масового обслуговування є багатоканальними, і, отже, моделі з n обслуговуючими каналами (де $n > 1$) представляють безперечний інтерес.



Процес масового обслуговування, описуваний даною моделлю, характеризується інтенсивністю вхідного потоку λ , при цьому паралельно може обслуговуватися не більш n клієнтів (заявок). Середня тривалість обслуговування однієї заявки дорівнює $1/\mu$. Вхідний і вихідний потоки є пуассонівськими. Режим функціонування того або іншого обслуговуючого каналу не впливає на режим функціонування інших обслуговуючих каналів системи, причому тривалість процедури обслуговування кожним з каналів є випадковою величиною, підлеглою експоненціальному закону розподілу. Кінцева мета використання n паралельно включених обслуговуючих каналів полягає в підвищенні (в порівнянні з одноканальною системою) швидкості обслуговування вимог за рахунок обслуговування одночасно n клієнтів.

Окремим видом СМО є системи масового обслуговування при наявності вхідного і вихідного потоків.

Це такі СМО, в яких є як вхідний потік, так і потік обслужених клієнтів. Досліджуються такі структури, в яких паралельно функціонують s каналів, так що одночасно можуть обслуговуватися відразу s клієнтів. При цьому всі обслуговуючі канали з погляду швидкодії передбачаються еквівалентними. Схематично така обслуговуюча система зображена на рисунку 1.5.

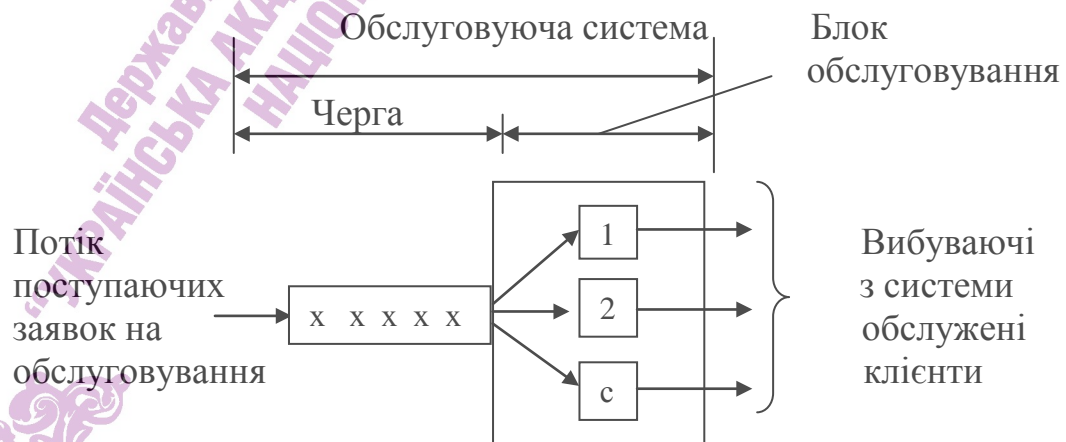


Рисунок 1.5 – Схеми роботи систем масового обслуговування при наявності вхідного і вихідного потоків

Помітимо, що в будь-який довільно вибраний момент часу всіх клієнтів, що знаходяться в системі, слід розділити на тих, хто знаходиться в черзі i , отже, чекає, коли його почнуть обслуговувати, і тих, хто вже обслуговується.

Введемо деякі позначення.

Позначення, які є найбільш відповідними для СМО з паралельно «включеними» каналами, давно вже уніфіковані і мають наступну структуру:

$$(a/b/c): (d/e/f),$$

де символи a, b, c, d, e і f асоційовані з конкретними найістотнішими елементами модельного представлення процесів масового обслуговування і інтерпретуються таким чином:

a – розподіл моментів надходжень заявок на обслуговування;

b – розподіл часу обслуговування (або вибуття обслужених клієнтів)

c – число паралельно функціонуючих вузлів обслуговування ($c=1,2,\dots,\infty$);

d – дисципліна черги;

e – максимальне число вимог, що допускаються в систему (число вимог в черзі плюс число вимог, прийнятих на обслуговування);

f – місткість джерела, що генерує заявки на обслуговування.

Для конкретизації a і b прийняті наступні стандартні позначення:

M – пуассоновський розподіл моментів надходження заявок на обслуговування або вибуття з системи обслужених клієнтів (або експоненціальний розподіл інтервалів часу між моментами послідовних надходжень або тривалостей обслуговування клієнтів);

D – фіксований (детермінований) інтервал часу між моментами послідовних надходжень в систему заявок на обслуговування або детермінована (фіксована) тривалість обслуговування;

E_k – розподіл Ерланга або гамма-розподіл інтервалів часу між моментами послідовних надходжень вимог в обслуговуючу систему або

тривалостей обслуговування (при цьому під до розуміється параметр розподілу);

GI – розподіл довільного виду моментів надходження в систему заявок на обслуговування (або інтервалів часу між послідовними надходженнями вимог);

G – розподіл довільного виду моментів вибуття з системи обслугованих клієнтів (або тривалостей обслуговування).

Кінцева мета аналізу систем і процесів масового обслуговування полягає в розробці критеріїв (або показників) ефективності функціонування СМО.

Отже нас цікавитимуть наступні операційні характеристики СМО:

P_n – вірогідність того, що в системі знаходиться n клієнтів (заявок на обслуговування);

L_s – середнє число клієнтів, що знаходяться в системі (заявок на обслуговування);

L_q – середнє число клієнтів черги на обслуговування;

W_s – середня тривалість перебування клієнта (заявки на обслуговування) в системі;

W_q – середня тривалість перебування клієнта (заявки на обслуговування) в черзі.

За визначенням

$$L_s = \sum_{n=0}^{\infty} n p_n \quad (1.4)$$

$$L_q = \sum_{n=c}^{\infty} (n - c) p_n \quad (1.5)$$

Між L_s і W_s (як і між L_q і W_q) існує строгий взаємозв'язок, так що, знаючи числові значення однієї з цих величин, можна легко знайти значення



іншої величини. Зокрема, якщо частота надходжень в систему заявок на обслуговування дорівнює λ (інтенсивність надходження вимог), то ми маємо

$$L_s = \lambda W_s \quad (1.6)$$

$$L_q = \lambda W_q \quad (1.7)$$

Приведені вище співвідношення справедливі і при таких припущеннях, що не накладають ніяких спеціальних обмежень ні на розподіл моментів послідовних надходжень вимог, ні на розподіл тривалостей обслуговування. Проте в тих випадках, коли частота надходжень заявок на обслуговування дорівнює λ , але не всі заявки мають нагоду потрапити в обслуговуючу систему (наприклад, через недостатньо велику місткість блоку очікування), співвідношення (1.3), (1.4) необхідно видозмінити шляхом такого нового визначення параметра λ , яке дозволило б врахувати вимоги, що тільки дійсно «допускаються» в систему. Тоді, вводячи в розгляд $\lambda_{E\Phi}$ – ефективну частоту надходжень, тобто кількість вимог, дійсно допущених в блок очікування обслуговуючої системи в одиницю часу, матимемо:

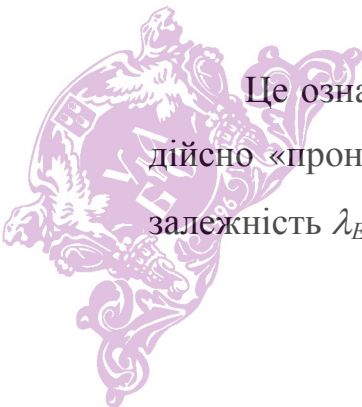
$$L_s = \lambda_{E\Phi} W_s \quad (1.8)$$

$$L_q = \lambda_{E\Phi} W_q \quad (1.9)$$

У загальному випадку

$$\lambda_{E\Phi} = \beta \lambda, \quad 0 < \beta < 1 \quad (1.10)$$

Це означає, що тільки частина поступаючих заявок на обслуговування дійсно «проникає» в систему. Але у будь-якому випадку можна встановити залежність $\lambda_{E\Phi}$ від L_s L_q таким чином.



За визначенням

$$\left[\begin{array}{c} \text{Середня трива-} \\ \text{лість перебування} \\ \text{в системі} \end{array} \right] = \left[\begin{array}{c} \text{Середня трива-} \\ \text{лість перебування} \\ \text{вимог у черзі} \end{array} \right] + \left[\begin{array}{c} \text{Середній час} \\ \text{обслуговування.} \end{array} \right]$$

Якщо середня швидкість обслуговування дорівнює μ і, отже, середня тривалість обслуговування дорівнює $1/\mu$, то справедливо наступне співвідношення:

$$W_s = W_q + \frac{1}{\mu} \quad (1.11)$$

Помножаючи ліву і праву частини цього співвідношення на λ , одержуємо:

$$L_s = L_q + \frac{\lambda}{\mu} \quad (1.12)$$

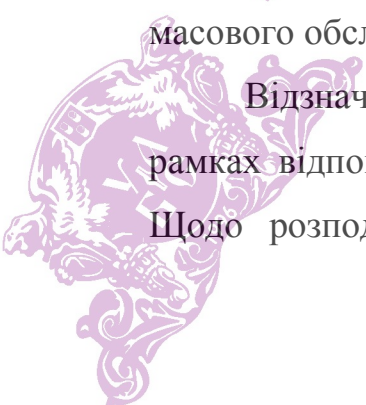
Останнє співвідношення залишається справедливим і в тому випадку, якщо λ замінити на $\lambda_{E\Phi}$. При цьому для $\lambda_{E\Phi}$ можна записати

$$\lambda_{E\Phi} = \mu(L_s - L_q) \quad (1.13)$$

При аналізі всіх моделей, що розглядаються нижче, основна увага буде зосереджена на отриманні формул для p_n , оскільки, знаючи p_n , неважко визначити значення всіх основних операційних характеристик процесу масового обслуговування, що цікавлять нас

Відзначимо, що в більшості випадків при обчисленні значень p_n в рамках відповідної математичної моделі особливих труднощів не виникає.

Щодо розподілів тривалостей очікування, то їх чисельна оцінка може



виявитися далеко не простою. Таким чином, в більшості випадків зручніше обчислювати W_S і W_Q через L_S і L_Q .

Основними типами моделей систем масового обслуговування є наступні.

Система масового обслуговування типу $(M/M/1):(GD/\infty/\infty)$.

У моделі $(M/M/1):(GD/\infty/\infty)$ є єдиний вузол обслуговування (обслуговуючий канал), а на місткість блоку очікування і місткість джерела вимог ніяких обмежень не накладається. Вхідний і вихідний потоки є пуассоновськими з параметрами λ і μ відповідно.

Система масового обслуговування типу $(M/M/1):(GD/N/\infty)$.

Різниця між моделлю типу $(M/M/1):(GD/N/\infty)$ і моделлю типу $(M/M/1):(GD/\infty/\infty)$ полягає тільки у тому, що кількість вимог, що допускаються в блок очікування обслуговуючої системи, дорівнює N . Це означає, що за наявності в системі N вимог жодна з додаткових заявок на обслуговування не може приєднуватися до черги в блоці очікування. В результаті ефективна частота надходжень вимог λ_{EF} для системи вказаного типу стають менше частоти λ , з якої заявки на обслуговування генеруються відповідним джерелом.

Система масового обслуговування типу $(M/M/c):(GD/\infty/\infty)$.

Процес масового обслуговування, описуваний такою моделлю, характеризується інтенсивністю вхідного потоку λ і тією обставиною, що паралельно обслуговуються може не більш c клієнтів. Середня тривалість обслуговування одного клієнта дорівнює $1/\mu$. Вхідний і вихідний потоки є пуассонівськими. Кінцева мета використання c паралельно працюючих каналів обслуговування полягає в підвищенні (в порівнянні з одноканальною системою) швидкості обслуговування вимог за рахунок обслуговування одночасно c клієнтів. Таким чином, якщо $n=c$, то інтенсивність вхідного (вихідного) потоку дорівнює μ . З другого боку, якщо $n < c$, то інтенсивність вхідного (вихідного) потоку дорівнює $n\mu < c\mu$ (оскільки при цьому зайнятими

обслуговуванням виявляться не всі обслуговуючі канали, а лише $n < c$) приладів). По суті, використання декількох обслуговуючих каналів еквівалентне використуванню одного обслуговуючого каналу, швидкодія якого варіюється, збільшуючись за наявності в системі n вимог рівно в n раз.

Таким чином, моделі теорії масового обслуговування дуже добре описують системи обслуговування клієнтів, подібною до яких є банкомат як система обслуговування власників пластикових карток. При моделюванні таких систем отримують результати, наближені до реальних. Точність даних, отриманих в процесі моделювання, може залежати від обраної моделі, а також від способу моделювання, яке може бути як аналітичним, так і статистичним. Проте, головним у моделях теорії масового обслуговування є те, що вони враховують стохастичний характер процесу обслуговування, а тому є найбільш реалістичними і дозволяють використовувати результати моделювання на практиці. Приклади таких моделей будуть детально розглянуті у другому розділі даної роботи.



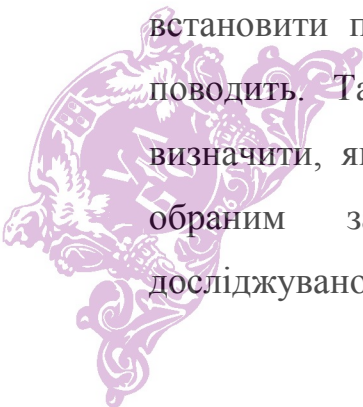
2 ІМІТАЦІЙНЕ МОДЕЛЮВАННЯ ЕФЕКТИВНОГО ОБСЛУГОВУВАННЯ БАНКОМАТІВ ТА ЇХ ОПТИМАЛЬНОГО РОЗМІЩЕННЯ

2.1 Моделі визначення строків та обсягів забезпеченості банкоматів грошовими коштами

Діяльність комерційного банку, як і будь-якого суб'єкту економіки, обумовлена багатьма чинниками, які мають випадкову, невизначену природу. Це стосується як внутрішніх процесів банку, так і зовнішніх умов, в яких він веде свою діяльність. Таким чином, робота банківської установи значно ускладнюється і потребує додаткових зусиль щодо планування і прогнозування. На допомогу банкам в цій ситуації приходять теорія ймовірностей, яка пропонує широкий набір інструментів для опису подібних нестабільних і невизначених явищ.

Досить складним, стохастичним процесом у банківській діяльності є обслуговування клієнтів, зокрема надання їм послуг через мережу банкоматів. Теорія ймовірностей досить детально розглядає питання обслуговування крізь призму теорії масового обслуговування. Найчастіше у даному розділі математики мова йде про такі заклади масового обслуговування, як магазини, аеропорти, майстерні тощо. У літературі досить рідко зустрічається тема банків як об'єктів подібних досліджень. Але обслуговування клієнтів банківськими установами безперечно може бути описане з позиції теорії масового обслуговування, адже потік клієнтів є аналогічним будь-якій установі і має подібні властивості.

У теорії ймовірностей для кожної величини, що досліджується, можна встановити певний закон розподілу, в залежності від того, як вона себе поводить. Таким чином, можна дослідити явище, яке нас цікавить, і визначити, яким законом розподілу його можна описати. Надалі згідно з обраним законом будуть визначатися основні характеристики досліджуваного об'єкту.



Знаючи закони розподілу основних величин, що використовуються при описі процесу обслуговування, ми можемо визначити їх значення за тих чи інших умов, змінювати деякі параметри об'єктів дослідження і зовнішнього середовища та знаходити оптимальні розв'язки поставленої задачі.

Отже теорія масового обслуговування зазвичай визначає потік клієнтів як пуассонівський (простіший), тобто такий, що має властивості стаціонарності, відсутності післядії та ординарності.

Властивість стаціонарності характеризується тим, що ймовірність виникнення k подій на будь-якому проміжку часу залежить тільки від числа k та від тривалості проміжку і не залежить від початку його відліку.

Властивість відсутності післядії полягає у тому, що ймовірність появи k подій на будь-якому проміжку часу не залежить від того, з'являлись чи ні події у моменти часу, що передували початку проміжку, який розглядається.

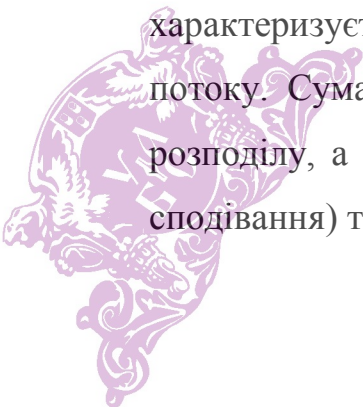
Властивість ординарності означає те, що поява двох або більше подій за досить малий проміжок часу практично неможлива.

Інтервал між подіями пуассонівського потоку має показниковий розподіл.

Таким чином, процес обслуговування клієнтів банку через банкомати також можна розглядати як стандартний пуассонівський потік, який має наступні основні характеристики:

- кількість клієнтів;
- сума операції;
- час обслуговування;
- накопичена сума за певний час.

У свою чергу, кількість клієнтів є випадковою величиною, яка характеризується ймовірностями конкретних значень та інтенсивністю потоку. Сума однієї операції та накопичена сума мають власні щільності розподілу, а також такі характеристики, як середнє значення (математичне сподівання) та квадратичне відхилення.



Таким чином, у процесі створення моделі, використовуючи інструментарій теорії ймовірностей та, зокрема, теорії масового обслуговування, ми можемо задавати вхідні та шукані параметри, варіювати умови задачі та знаходити необхідні розв'язки.

Однією з моделей, які базуються на теорії ймовірностей, є модель, розроблена російським економістом Васіним Н.С.

При вирішенні задачі своєчасного підкріплення банкоматів готівкою необхідно враховувати, що потік клієнтів і сума грошей кожної операції, здійснюваної клієнтом, є випадковими. Випадковим є і фінансовий потік, що формується у результаті виконання цих операцій. Для розв'язання стохастичної невизначеності, спричиненої цими фактами, необхідно використовувати теоретико-ймовірнісну модель, яка дозволяє прогнозувати, керувати ризиком зниження якості обслуговування банкоматної мережі, обумовленим несвоечасним підкріпленням банкоматів готівкою.

Розглянемо поставлену задачу при наступних вихідних припущеннях. Нехай потік клієнтів є стаціонарним пуассонівським процесом [6]. Отже число клієнтів за даний час t N_t є випадковою величиною з ймовірностями конкретних значень n

$$P(N_t = n) = P_t(n) = \frac{(\lambda \cdot t)^n}{n!} \cdot e^{-\lambda \cdot t}, \quad (2.1)$$

$$n = 0, 1, 2, \dots$$

де λ – інтенсивність потоку клієнтів.

Середнє число клієнтів, що поступили за час t визначається за формулою:

$$\bar{N}_t = \lambda \cdot t \quad (2.2)$$

при квадратичному відхиленні



$$\sigma_{N_t} = \sqrt{\lambda \cdot t} \quad (2.3)$$

Сума операції G як випадкова величина характеризується щільністю розподілу $f(g)$ із середнім значенням

$$\bar{G} = \int_0^{\infty} f(g)gdg \quad (2.4)$$

і квадратичним відхиленням

$$\sigma_G = \sqrt{\int_0^{\infty} f(g)g^2dg - \bar{G}^2} \quad (2.5)$$

Накоплена сума за час t S_t також є випадковою величиною з щільністю розподілу

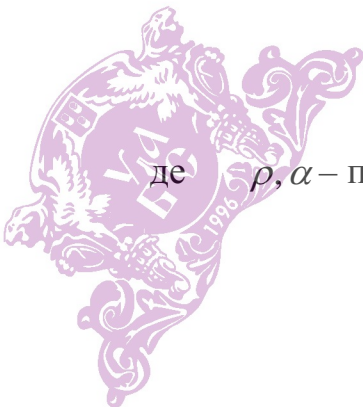
$$f_t(g) = \sum_{n=0}^{\infty} P_t(n) \cdot f^{*n}(g) = e^{-\lambda \cdot t} \sum_{n=0}^{\infty} \frac{(\lambda \cdot t)^n}{n!} \cdot f^{*n}(g) \quad (2.6)$$

де f^{*n} – щільність розподілу суми n операцій S_n .

У якості вичерпної теоретико-ймовірності характеристики суми одної банкоматної операції можна розглядати гама-розподіл з щільністю

$$f(g) = \frac{1}{\rho \Gamma(\alpha)} \left(\frac{g}{\rho} \right)^{\alpha-1} \cdot \exp\left(-\frac{g}{\rho} \right), \quad g \geq 0 \quad (2.7)$$

де ρ, α – параметри розподілу:



$$\bar{G} = \rho \cdot \alpha \quad (2.8)$$

$$\sigma_G = \rho \cdot \sqrt{\alpha} \quad (2.9)$$

Виходячи з цих припущень, розглянемо задачу визначення інтервалу часу T_s , за який накоплена сума S_t вперше перевищить заданий рівень S , починаючи з моменту $t=0$, коли $S_t=0$. Вважатимемо, що потік клієнтів пуассонівський з інтенсивністю λ і сума операції G як випадкова величина має щільність $f(g)$ і $G \geq 0$.

Якщо позначити p_n ймовірність того, що на n -м клієнті S_t перевищить S , то час настання цієї події T_n як випадкова величина має щільність розподілу

$$\varphi_n(t) = \frac{\lambda}{(n-1)!} \cdot (\lambda \cdot t)^{n-1} \cdot e^{-\lambda t}, \quad t \geq 0 \quad (2.10)$$

який називається спеціальним розподілом Ерланга. Цей розподіл є окремим випадком гама-розподілу (2.7) при цілому значенні параметру α . Оскільки інтервал між подіями пуассонівського потоку має показниковий розподіл, то щільність (2.10) отримуємо як n -кратну згортку показникового розподілу $\varphi(t) = \lambda \cdot e^{-\lambda t}$. Це впливає з того, що

$$T_n = \sum_{i=1}^n \Delta t_i \quad (2.11)$$

де Δt_i – інтервал між моментами надходження $(i-1)$ -го і i -го клієнтів, що має показниковий розподіл.

Із урахуванням цього час T_s має щільність розподілу

$$\varphi_s(t) = \sum_{n=1}^{\infty} p_n \cdot \varphi_n(t) \quad (2.12)$$



Залишається тільки визначити ймовірність $p_n(S)$. Якщо визначити $F_n(g)$ як функцію розподілу накопиченої суми від n клієнтів S_n , тобто

$$F_n(g) = \int_0^g f^{*n}(g) dg \quad (2.13)$$

то

$$p_n(S) = F_{n-1}(S) - F_n(S), \quad F_0(S) = 1 \quad (2.14)$$

Цей вираз є наслідком того, що $F_n(S)$ (ймовірність того, що $S_n \leq S$) дорівнює ймовірності того, що $N_s \geq n$, де N_s – номер клієнта, на якому вперше сума S_n перевищила S . Формально це можна записати так:

$$F_n(S) = P(N_s \geq n) \quad (2.15)$$

Так як $p_n(S) = P(N_s = n) = P(N_s > n-1) - P(N_s > n)$, то з урахуванням (2.15), одержуємо (2.14).

Величина N_s являє самостійний практичний інтерес. Наприклад, якщо S – запас грошей в банкоматі, то N_s – номер першого клієнта, якому не вистачило грошей, а $N_s = \bar{N}'_s - 1$ – номер останнього клієнта, якому вистачило грошей. Величина, аналогічна N_s , в теорії відновлення називається числом відновлень за час, рівний S . Скористуємося тут деякими відомими результатами. Математичне сподівання \bar{N}'_s визначається з наступного інтегрального рівняння:

$$\bar{N}'_s = F(S) + \int_0^S \bar{N}'_{s-g} \cdot f(g) dg \quad (2.16)$$

де



$$F(S) = \int_0^S f(g) dg \quad (2.17)$$

Якщо $f(g)$ має показниковий розподіл, тобто

$$f(g) = \frac{1}{\bar{G} \cdot \exp(-g/\bar{G})} \quad (2.18)$$

то $\bar{N}'_s = S/\bar{G}$. В загальному випадку наведене інтегральне рівняння не має явного рішення, але є просте асимптотичне розв'язання:

$$\bar{N}'_s \approx S/\bar{G} + 0.5(K_G - 1), \quad K_G = \sigma_G/\bar{G} \quad (2.19)$$

з якого одержуємо, що

$$\bar{N}_s = \bar{N}'_s + 1 \approx S/\bar{G} + 0.5(K_G + 1) \quad (2.20)$$

Відомо також асимптотичне рішення для дисперсії

$$\sigma_{N_s}^2 \approx S \cdot K_G / \bar{G} + (1/12 + 5K_G^4/4 - 2A_G \cdot K_G^3/3) \quad (2.21)$$

де A_G – показник асиметрії щільності розподілу $f(g)$, рівний відношенню третього центрального моменту до кубу квадратичного відхилення суми операцій G . Для показникового розподілу (2.18) $A_G = 2$, а для гама-розподілу (2.7) $A_G = 2K_G$. Наведені асимптотичні розв'язки є тим точнішими, чим більше S , а у випадку показникового розподілу вони дають точне рішення для будь-яких S .

З урахуванням виразів (2.10), (2.14) одержуємо, що щільність розподілу T_s має вигляд:

$$\varphi_s(t) = \lambda e^{-\lambda t} \cdot \sum_{n=1}^{\infty} [F_{n-1}(S) - F_n(S)] \cdot \frac{(\lambda t)^{n-1}}{(n-1)!} \quad (2.22)$$

Якщо сума операції G має показниковий розподіл (2.18), то одержуємо більш простий явний вираз:

$$\varphi_s(t) = \lambda \cdot \exp(-\lambda t - S/\bar{G}) I_0(2\sqrt{\lambda t S/\bar{G}}) \quad (2.23)$$

де $I_0()$ – модифікована функція Беселя нульового порядку. Щільності типу (2.23) називають беселевими.

Побудуємо графіки щільностей $\varphi_s(t)$ при різних значеннях S .

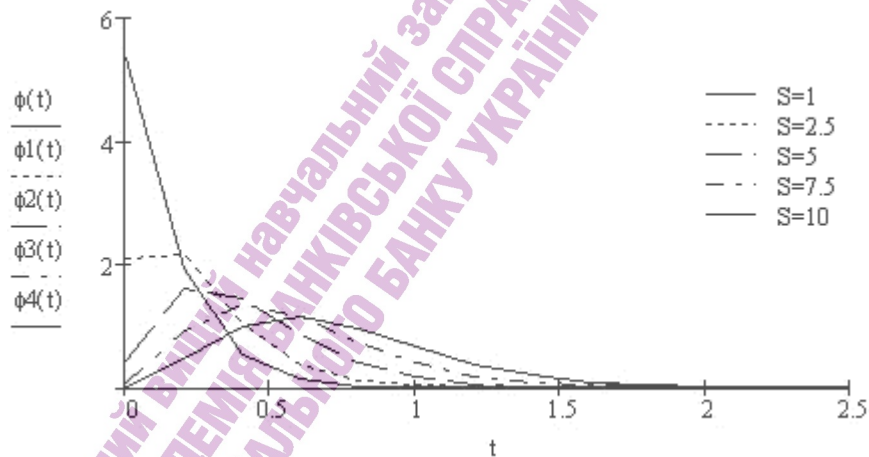


Рисунок 2.1– Графік щільностей $\varphi_s(t)$ розподілу часу T_s першого досягнення накопиченої суми значення, рівного або більшого S

Математичне сподівання і дисперсію T_s одержимо з використанням перетворення Лапласа, але по змінній t .

$$\varphi_s^\circ(\omega) = \int_0^{\infty} \varphi_s(t) e^{-\omega t} dt = \sum_{n=1}^{\infty} p_n(s) \cdot \left(\frac{\lambda}{\lambda + \omega} \right)^n \quad (2.24)$$

$$\bar{T}_s = - \left. \frac{d\varphi_s^\circ(\omega)}{d\omega} \right|_{\omega=0} = \frac{1}{\lambda} \sum_{n=1}^{\infty} p_n(S) \cdot n = \bar{N}_s / \lambda \quad (2.25)$$



$$\sigma_{T_s}^2 = \left. \frac{d^2 \varphi_s^\circ(\omega)}{d\omega^2} \right|_{\omega=0} - \bar{T}_s^2 = \frac{1}{\lambda^2} \sum_{n=1}^{\infty} p_n(S) n(n+1) - \bar{T}_s^2 = \frac{1}{\lambda^2} (\bar{N}_s + \sigma_{N_s}^2) \quad (2.26)$$

Для достатньо великих S час T_s має асимптотично нормальний розподіл, тобто

$$\varphi_s(t) \approx \frac{1}{\sqrt{2\pi\sigma_{T_s}}} \exp\left[-\frac{(t - \bar{T}_s)^2}{2\sigma_{T_s}^2}\right] \quad (2.27)$$

де \bar{T}_s та σ_{T_s} розраховуються за формулами (2.10), (2.12).

Якщо G має показниковий розподіл, то

$$\bar{T}_s = \frac{1}{\lambda} \left(\frac{S}{G} + 1 \right) \quad (2.28)$$

$$\sigma_{T_s}^2 = \frac{1}{\lambda^2} \left(\frac{2S}{G} + 1 \right) \quad (2.29)$$

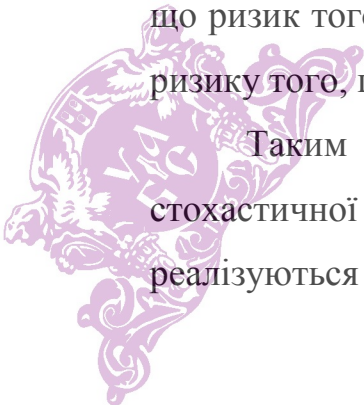
і це є точні рішення.

Між $f_t(g)$ і $\varphi_s(t)$ існує однозначний зв'язок, а саме

$$\int_0^T \varphi_s(t) dt = \int_S^\infty f_T(g) dg \quad (2.30)$$

який випливає з факту, що ймовірність того, що $T_s \leq T$ дорівнює ймовірності того, що $S_T \geq S$. Якщо S_t – витрати грошей, то співвідношення (2.30) означає, що ризик того, що витрати за час T перевищать задану величину S , дорівнює ризику того, що рівень витрат S буде досягнутий не пізніше T .

Таким чином, наведений математичний апарат розкриває зміст стохастичної невизначеності, яка супроводжує бізнес-процеси, що реалізуються в банкоматних системах. На його основі можна будувати



систему інформаційно-аналітичної підтримки інкасаційного обслуговування банкоматів.

Проте, використання аналітичних моделей, використовуваних при теоретико-ймовірнісному підході, містить деякі припущення і асимптотичні рішення, а у ряді випадків аналітичний розв'язок взагалі є неможливим. Тому для отримання необхідних результатів можна використовувати методи статистичного моделювання, які до того ж представляють собою комп'ютерний експеримент, що дозволяє перевірити результати аналітичного моделювання.

При моделюванні потоку клієнтів будемо вважати, як і за теорією масового обслуговування, що потік пуассонівський, але інтенсивність потоку залежить від часу, тоді кількість клієнтів, що поступають у середньому за час t ,

$$\Lambda(t) = \int_0^t \lambda(t) dt \quad (2.31)$$

Ймовірність того, що за інтервал часу від t_1 до t_2 поступить n клієнтів

$$P_n(t_1, t_2) = \exp\{-[\Lambda(t_2) - \Lambda(t_1)]\} \frac{[\Lambda(t_2) - \Lambda(t_1)]^n}{n!} \quad (2.32)$$

Якщо врахувати (2.31), то

$$P_n(t_1, t_2) = \exp\left\{-\int_{t_1}^{t_2} \lambda(t) dt\right\} \frac{\left[\int_{t_1}^{t_2} \lambda(t) dt\right]^n}{n!} \quad 2.33$$

Для нестационарного потоку Пуассона використовуємо метод введення операційного часу. Ідея метода полягає в тому, що робочий час банкомату



протягом доби t замінюється операційним часом t' таким, що у рівні проміжки часу надходило б у середньому рівна кількість клієнтів. Тобто нестационарний процес Пуассона у звичайному часу стає стаціонарним у операційному часі і його можна моделювати при відповідному $\lambda = \lambda'$. Отримані таким чином моменти надходження клієнтів в операційному часі t' перераховується затим у моменти звичайного часу t . При цьому необхідно визначити операційну інтенсивність потоку клієнтів λ' і алгоритм переходу від операційного часу t' до звичайного часу t .

Розглянемо інтервал часу з початку до кінця доби, тобто $t_1=0$, а $t_2=T$, де $T=24$ години. Тоді кількість клієнтів, що надійшли протягом доби

$$\Lambda(T) = \int_0^T \lambda(t) dt \quad (2.34)$$

Операційний час t' оберемо так, щоб за $t=0$ і $t=T$ операційний час t' мало ті ж значення, а інтенсивність потоку

$$\lambda' = \frac{1}{T} \int_0^T \lambda(t) dt \quad (2.35)$$

де λ' – це середня інтенсивність потоку клієнтів протягом доби.

Встановимо тепер зв'язок між t і t' . Протягом доби кількість клієнтів, які надійшли до моменту t $\Lambda(t)$, співпадала з кількістю клієнтів у операційному часі до моменту t' . Тобто

$$\Lambda(t) = \lambda' t' \quad (2.36)$$

Звідси отримуємо, що



$$t' = \frac{\Lambda(t)}{\lambda'} = \frac{1}{\lambda'} \int_0^t \lambda(t) dt \quad (2.37)$$

Зворотна залежність t від t' , яку позначимо, отримується шляхом вирішення рівняння (2.37) відносно t .

У загальному випадку явний вираз для $\eta(t')$ отримати не вдасться, тому рівняння (2.37) доводиться розв'язувати чисельно. Маємо наступний алгоритм. З шагом Δ обчислюємо інтеграл

$$I_i = \frac{1}{\lambda'} \int_0^{i\Delta} \lambda(t) dt \quad (2.38)$$

для $i=1, 2, 3, \dots$ до значення $i=k$, коли вперше I_i перевищить t' . Після цього методом лінійного інтерполювання отримуємо, що

$$I_i = \eta(t') = \Delta t \left(k - \frac{I_k - t'}{I_k - I_{k-1}} \right) \quad (2.39)$$

Для підвищення швидкодії алгоритму розрахунку за формулою (2.39) доцільно завчасно обчислити інтеграл (2.38) для усіх значень i та зберігати його у вигляді масиву в оперативній пам'яті комп'ютера.

Якщо t_j – момент надходження клієнта за звичайним часом, а t_j' – за операційним часом, то інтервал часу до надходження $(j+1)$ -го клієнта за операційним часом

$$\Delta t' = -\frac{1}{\lambda'} \ln(\xi) \quad (2.40)$$

а момент надходження цього клієнта за операційним часом



$$t'_{j+1} = t'_j + \Delta t' \quad (2.41)$$

Момент надходження цього клієнта за звичайним часом

$$t_{j+1} = \eta(t'_{j+1}) \quad (2.42)$$

де функція η розраховується за формулою (2.39).

Графічна ілюстрація таких перетворень представлена на рисунку 2.2.

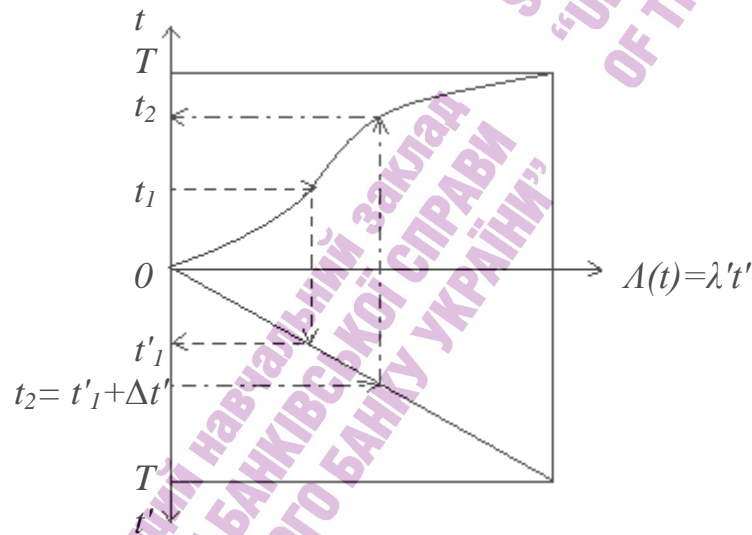


Рисунок 2.2 – Графічна ілюстрація переходу від робочого часу до операційного і навпаки.

Таким чином, отримано алгоритм розрахунку моментів надходження клієнтів у нестационарному потоці клієнтів. Ітерації закінчуються, якщо $|t_i - t_{i-1}|$ стає менше допустимої помилки ε , наприклад $\varepsilon = 0,0001$. За нульового наближення $t = 0$, а $t = T$, тому отримуємо, що $t_0 = t'$.

Як видно з результатів аналізу статистики банкоматних операцій, тривалість обслуговування клієнтів τ досить добре описується логнормальним розподілом з щільністю

$$\varphi(t) = \frac{1}{\sqrt{2\pi\alpha_\tau t}} \exp\left[-\frac{(\ln t - \ln \rho_\tau)^2}{2\alpha_\tau^2}\right] \quad (2.43)$$

де α_τ, ρ_τ – параметри розподілу, які оцінюються за дослідними даними. Алгоритм моделювання випадкової величини з таким розподілом полягає у тому, що тривалість обслуговування кожного клієнта розраховується за формулою

$$\tau = \exp\left[\rho_\tau + \alpha_\tau \left(\sum_{i=1}^{12} \gamma_i - 6\right)\right] \quad (2.44)$$

де γ_i – 12 випадкових чисел, рівномірно розподілених у інтервалі 0...1, генеруємих стандартною підпрограмою Паскаля `random`.

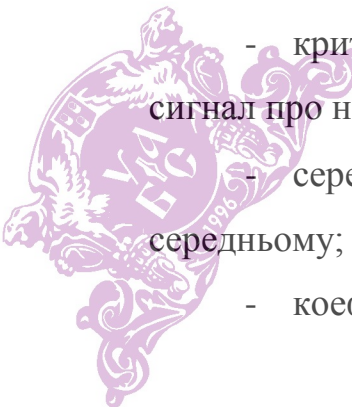
У результаті виконання операції по обслуговуванню клієнт отримує суму грошей G , яка як випадкова величина добре описується гамма-розподілом з щільністю

$$f(g) = \frac{1}{\rho_G \Gamma(\alpha_G)} \left(\frac{g}{\rho_G}\right)^{\alpha_G-1} \exp\left(-\frac{g}{\rho_G}\right) \quad (2.45)$$

де ρ_G, α_G – параметри розподілу, які також оцінюються за дослідними даними.

Вихідними даними для моделювання функціонування банкоматної системи є:

- сума завантаження банкомату Z_0 ;
- критична сума грошей в банкоматі Z_k , при досягненні якої подається сигнал про необхідність завантаження банкомату;
- середня сума операції \bar{G} , тобто сума, яку запитує клієнт у середньому;
- коефіцієнт варіації суми G K_G ;



- середня тривалість операції по обслуговуванню клієнта $\bar{\tau}$;
- коефіцієнт варіації тривалості операції K_{τ} ;
- середня тривалість операції по обслуговуванню і завантаженню банкомата \bar{t}_s ;
- мінімальний інтервал часу з моменту подання сигналу про необхідність завантаження банкомату до початку операції завантаження \bar{t}_z ;
- тривалість циклу, що моделюється, у потоці клієнтів (число робочих днів у тижні) N_c ;
- кількість клієнтів за добами циклу, що моделюється N_j ;
- параметри розподілу клієнтів за часом доби;
- число реалізацій циклу, що моделюється N_r .

Стан банкомату у довільний момент часу доби t характеризується наступними параметрами:

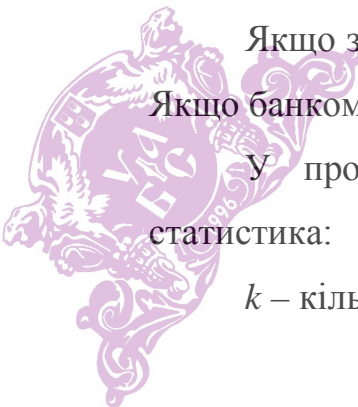
- поточним запасом грошей у банкоматі Z ;
- поточною чергою клієнтів, включно з клієнтом, що обслуговується O_c ;
- інтервалом часу з поточного моменту до моменту приходу наступного клієнта $\Delta\tau$;
- інтервалом часу з поточного моменту до моменту закінчення операції по обслуговуванню наступного клієнта Δt ;
- інтервалом часу з поточного моменту до моменту початку завантаження банкомату Δt_z ;
- інтервалом часу з поточного моменту при завантаженні банкомату до закінчення цієї операції $\Delta t'_z$;
- індексом стану банкомату si .

Якщо запас грошей у банкоматі $Z > Z_k$, то $si = 0$. Якщо $Z \leq Z_k$, то $si = 1$.

Якщо банкомат знаходиться у стані обслуговування і завантаження, то $si = 2$.

У процесі моделювання роботи банкомату збирається наступна статистика:

k – кількість клієнтів, що надійшли;



k_0 – кількість клієнтів, що було обслужено;

k^- – кількість клієнтів, які не отримали гроші через те, що сума, яка була запитана, менше залишку грошей у банкоматі ($G < Z$) або банкомат було заблоковано у зв'язку з його завантаженням і обслуговуванням;

$S_{Oч}$ – статистика для розрахунку середньої довжини черги;

S_{Oc} – статистика для розрахунку середньої величини залишку коштів при інкасації;

t_{H1} – накоплений час перебування банкомату у стані очікування клієнтів, тобто стані, коли $Oч = 0$;

t_{H2} – накоплений час перебування банкомату у стані очікування завантаження за відсутності грошей у ньому ($Z = 0$);

t_{H3} – накоплений час перебування банкомату у стані обслуговування і завантаження ($si = 2$);

N_Z – накоплена кількість завантажень банкомата;

S_z – статистика для розрахунку середнього запасу грошей в банкоматі.

Після закінчення моделювання за зібраною статистикою розраховуються підсумкові показники роботи банкомата як системи масового обслуговування.

Середня кількість клієнтів, що надійшли за день, розраховується за формулою

$$\bar{k} = \frac{k}{N_{\text{дн}}} \quad (2.46)$$

де $N_{\text{дн}}$ – число днів роботи банкомату в процесі моделювання, що розраховується за формулою

$$N_{\text{дн}} = N_c N_r \quad (2.47)$$

Середня кількість обслужених за добу клієнтів розраховується



$$\bar{k} = \frac{k_0}{N_{\text{дн}}} \quad (2.48)$$

Середнє число клієнтів, що отримали відмову в обслуговуванні складає

$$\bar{k}^- = \frac{k^-}{N_{\text{дн}}} \quad (2.49)$$

Середній період завантажень у днях

$$\bar{T}_z = \frac{N_{\text{дн}}}{N_z} \quad (2.50)$$

Середня сума залишку при інкасації банкомату

$$\bar{Z}_{oc} = \frac{S_{oc}}{N_z} \quad (2.51)$$

Середня довжина черги клієнтів до банкомату

$$\bar{O}_ч = \frac{S_{oc}}{t_\phi} \quad (2.52)$$

де t_ϕ – тривалість інтервалу моделювання у годинах:

$$t_\phi = N_{\text{дн}} 24 \quad (2.53)$$

Середній запас коштів у банкоматі складає



$$\bar{Z} = \frac{S_z}{t_\phi} \quad (2.54)$$

Коефіцієнт простою банкомату через очікування клієнтів

$$K_{H1} = \frac{t_{H1}}{t_\phi} \quad (2.55)$$

Коефіцієнт простою через закінчення коштів у банкоматі

$$K_{H2} = \frac{t_{H2}}{t_\phi} \quad (2.56)$$

Коефіцієнт простою через обслуговування і завантаження банкомату

$$K_{H3} = \frac{t_{H3}}{t_\phi} \quad (2.57)$$

Коефіцієнт використання банкомату розраховується

$$K_u = 1 - K_{H1} - K_{H2} - K_{H3} \quad (2.58)$$

На рисунку 2.3 наведено блок-схема алгоритму статистичного моделювання функціонування банкомату.

У блоці 1 із попередньо підготовленого файлу береться інформація про вихідні дані.





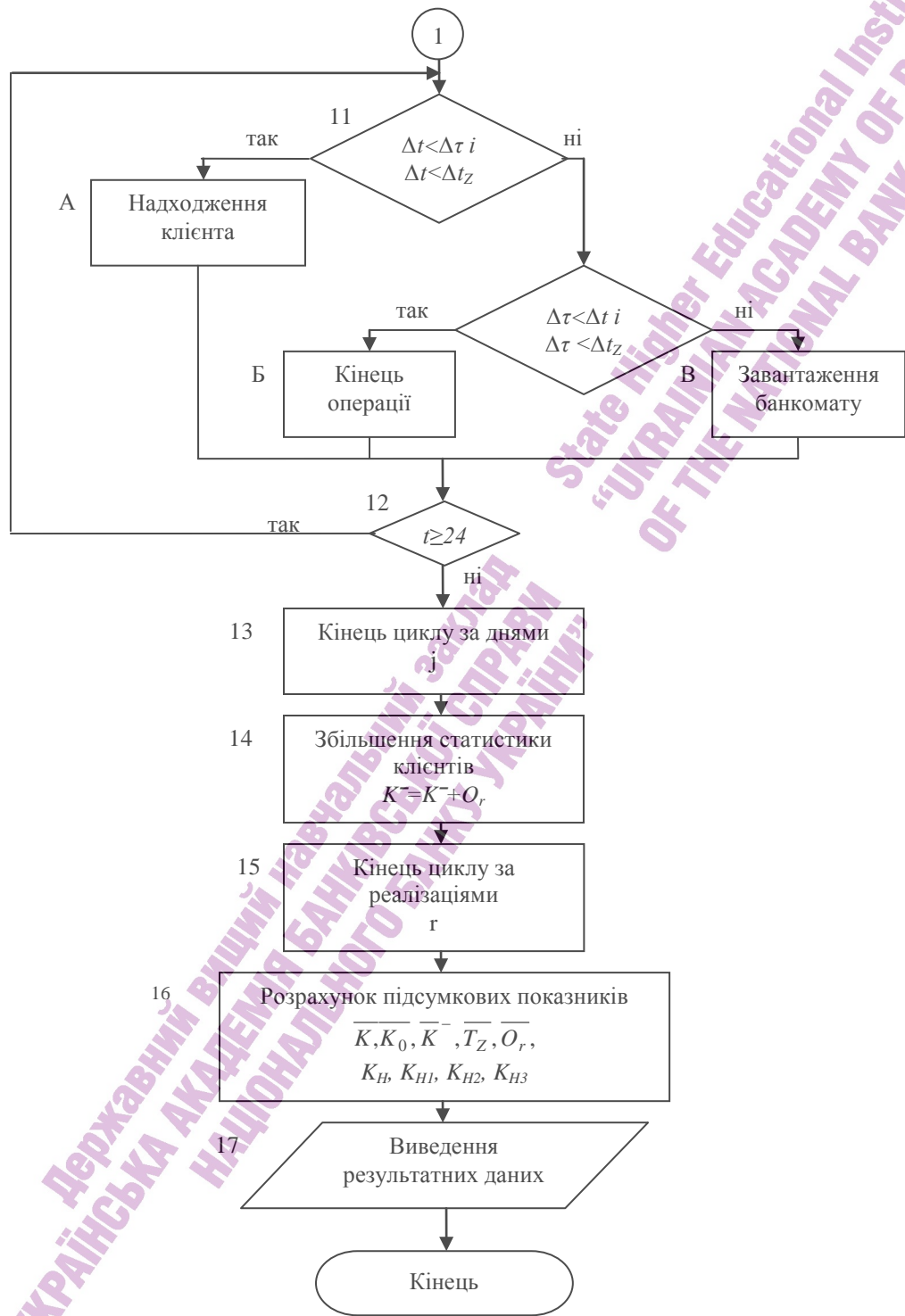
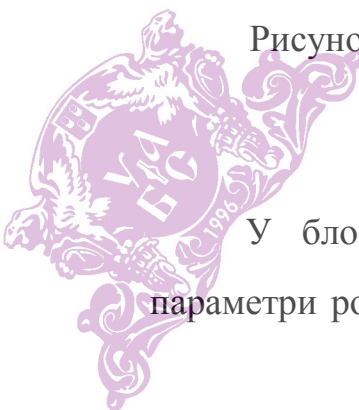


Рисунок 2.3 – Блок-схема алгоритму статистичного моделювання функціонування банкомату

У блоці 2 за раніше виведеними залежностями розраховуються параметри розподілу тривалості обслуговування ρ_τ , a_τ за τ і K_τ параметри



розподілу суми операції ρ_G , α_G за \bar{G} і K_G . Ці параметри потім використовуються для генерування випадкової тривалості операції τ і суми операції G .

У блоці 3 розраховуються накопичені значення частки клієнтів, що надходять, за годинами доби F_i . Якщо P_i – частки клієнтів за годинами доби ($i=1\dots 24$), то

$$F_i = \sum_{j=1}^i P_j \quad 2.59$$

Отже $F_{24} = 1$.

У блоці 4 усі описані вище статистики прирівнюються нулю.

У блоці 5 дається початкове завантаження банкомату і визначається середня інтенсивність потоку клієнтів у перший день моделювання.

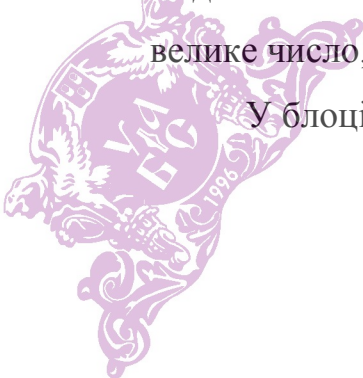
У блоці 6 Організується цикл за числом реалізацій N_r . Це число бажано брати якомога більшим.

У блоці 7 задаються початкові значення параметрів стану банкомату:

- черги $O_c = 0$;
- індексу стану $s_i = 0$;
- часу до закінчення операції $\Delta\tau$
- часу до початку завантаження $\Delta t_z = \infty$.

Нескінченність означає, що дана подія не може відбутися у даній ситуації. Дійсно, якщо черга нульова, то і операція по обслуговуванню неможлива. Якщо $Z = Z_0$, що більше Z_k , то час до завантаження ще невизначено, що рівносильне заданню нескінченності цього часу. При моделюванні роль нескінченності може відігравати будь-яке достатньо велике число, наприклад 10^{30} .

У блоці 8 організується цикл за днями тижня.



У блоці 9 визначаються початкові значення поточного часу в межах доби, середня інтенсивність потоку клієнтів для відповідного дня тижня і годин доби.

У блоці 10 визначається час до приходу наступного клієнта.

У блоці 11 визначається час до найближчої події. При цьому відслідковуються наступні події:

- надходження наступного клієнта;
- закінчення операції по обслуговуванню клієнта;
- початок завантаження банкомату;
- закінчення завантаження банкомату.

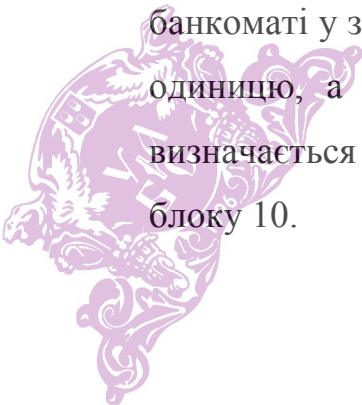
У залежності від того, яка подія відбудеться раніше, відбувається розгалуження алгоритму.

У блоці А відбувається обробка ситуації у зв'язком з надходженням наступного клієнта. При цьому:

- число клієнтів, що надійшли k , збільшується на один;
- статистика S_{oc} збільшується на $Oч\Delta t$;
- поточний час t збільшується на Δt .

Якщо у межах інтервалу Δt черга була рівна нулю, то статистика t_{H1} збільшується на Δt , і клієнт одразу приймається на обслуговування, а час до закінчення його обслуговування $\Delta \tau$ визначається генеруванням випадкової довжини обслуговування, розподіленої за логнормальним законом. Якщо у черзі були клієнти, то черга збільшується на одиницю, а час до закінчення обслуговування клієнта $\Delta \tau$ зменшується на Δt .

Якщо сигнал про необхідність завантаження було подано ($si = 1$), то Δt_z також зменшується на Δt . Якщо клієнт надійшов при заблокованому банкоматі у зв'язку з його завантаженням ($si = 2$), то число k^- збільшується на одиницю, а t_{H3} збільшується на Δt . Після всіх відмічених перерахувань визначається новий інтервал часу до приходу наступного клієнта аналогічно блоку 10.



У блоці Б обробляється ситуація закінчення операції по обслуговуванню клієнта. При цьому:

- статистика S_{oc} збільшується на $Oч\Delta\tau$;
- черга зменшується на одиницю;
- інтервал Δt зменшується на $\Delta\tau$;
- поточний час t збільшується на $\Delta\tau$;
- розмір суми G , яку запитує клієнт, генерується у відповідності гама-розподілу.

Якщо виявляється, що $G > Z$, то обслуговування закінчується відмовою у видачі коштів, статистика k^- збільшується на одиницю. Якщо у черзі є клієнти, то на обслуговування береться наступний клієнт, визначається час його обслуговування $\Delta\tau$, якщо клієнтів немає, то $\Delta\tau = \infty$. Якщо $G < Z$, то статистика k_o збільшується на одиницю, запас Z зменшується на G . Якщо клієнти є, то на обслуговування надходить наступний клієнт і визначається час до закінчення його обслуговування $\Delta\tau$, як і у попередній ситуації. Аналогічно, якщо клієнтів немає, то $\Delta\tau = \infty$. Якщо після видачі грошей запас Z стане менше або рівний Z_k , то подається сигнал про необхідність завантаження банкомату ($si = 1$) і визначається інтервал часу до завантаження Δt_z . При цьому, якщо момент завантаження приходить на неробочий час служби, яка виконує завантаження, то цей час \bar{t}_z збільшується таким чином, щоб це завантаження відбулася у робочий час, наприклад, на час з 7 до 19 години. Інтервал цього часу необхідно задавати у програмі. Якщо сигнал про необхідність завантаження було подано раніше, то Δt_z зменшується на $\Delta\tau$.

У блоці В обробляється ситуація настання моменту початку і закінчення завантаження банкомату. При цьому статистика S_{oc} збільшується на $Oч\Delta t_z$, поточний час t збільшується на Δt_z , а Δt зменшується на Δt_z , $\Delta\tau = \infty$. Якщо $si = 1$, то відбувається інкасація банкомату, тобто вилучаються залишки і виконується завантаження і технічне обслуговування банкомату, а статистика t_{H2} збільшується на Δt_z . Час до закінчення цієї операції дорівнює $\Delta t'_z = \bar{t}_s$, індекс стану $si = 2$, статистика залишків S_{oc} збільшується на Z , а

поточний запас Z стає рівним нулю. Статистика k збільшується на число клієнтів у черзі, тобто на O_c , а якщо черга була нульовою, то статистика t_{H1} збільшується на Δt_z . Якщо $si = 2$ (завантаження закінчено), то статистика t_{H3} збільшується на Δt_z , $Z = Z_0$, $\Delta t_z = \infty$, число завантажень N_z збільшується на одиницю.

У блоці 12 перевіряється настання кінця доби. Якщо він не настав, то виконується перехід до блоку 11 і визначається, яка подія буде наступною. Якщо кінець доби настав, то здійснюється перехід до наступної доби, при цьому поточний час t починає відраховуватися з початку наступної доби.

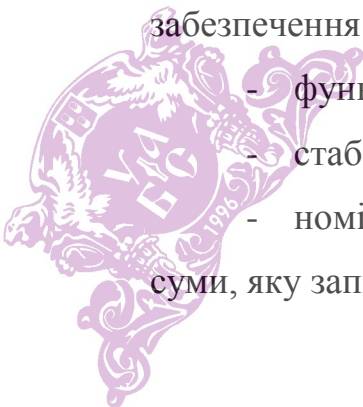
Після завершення циклу за кількістю реалізацій у блоці 16 за наведеними вище формулами розраховуються підсумкові показники роботи банкомата, а у блоці 17 виконується виведення результатів.

На основі аналізу отриманих результатів можна, варіюючи параметри процесу обслуговування банкоматних систем і застосовуючи статистичне моделювання, здійснити оптимізацію цього процесу за обраними критеріями, що буде сприяти підвищенню ефективності функціонування банкоматів і підвищить якість банківських послуг.

2.2 Визначення оптимального розміщення банкоматної мережі з використанням карти інтенсивності потоку клієнтів

У процесі формування і обслуговування банкоматної мережі слід враховувати той факт, що інтенсивність використання банкоматів є різною і залежить від ряду факторів. Серед них можна виділити наступні:

- зручність моделі банкомату та інтерфейсу його програмного забезпечення для користувачів;
- функціональність банкомату;
- стабільність роботи банкомату;
- номінал купюр, що містить банкомат, та спосіб формування з них суми, яку запитує клієнт;



- цінова політика банкомата щодо обслуговування різних пластикових карток та інші.

Проте, головним фактором, що впливає на інтенсивність користування банкоматом, є його розміщення.

Традиційно банки розміщують свої банкомати у місцях, де потік клієнтів може бути найбільш інтенсивним. Наприклад:

- торгівельні центри (великі магазини, ринки, сукупність дрібних магазинів);
- розважальні центри;
- підприємства і установи, які беруть участь у зарплатних проектах;
- вокзали;
- лікарні;
- великі житлові масиви, які мають у наявності один чи декілька магазинів тощо.

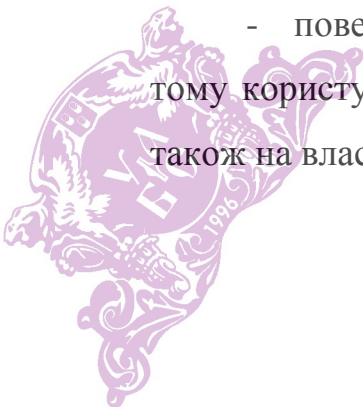
Отже, це такі місця, де у певний час зростає інтенсивність потоку користувачів банкоматів. Тому необхідно розрахувати інтенсивність цього потоку у визначений час на визначеній території.

Для цього перш за все потрібно скласти детальну карту міста, для якої знадобиться наступна інформація:

- щільність проживання населення;
- соціально-економічна інфраструктура міста;
- інформація про зарплатні проекти;
- карта мережі банкоматів у місті.

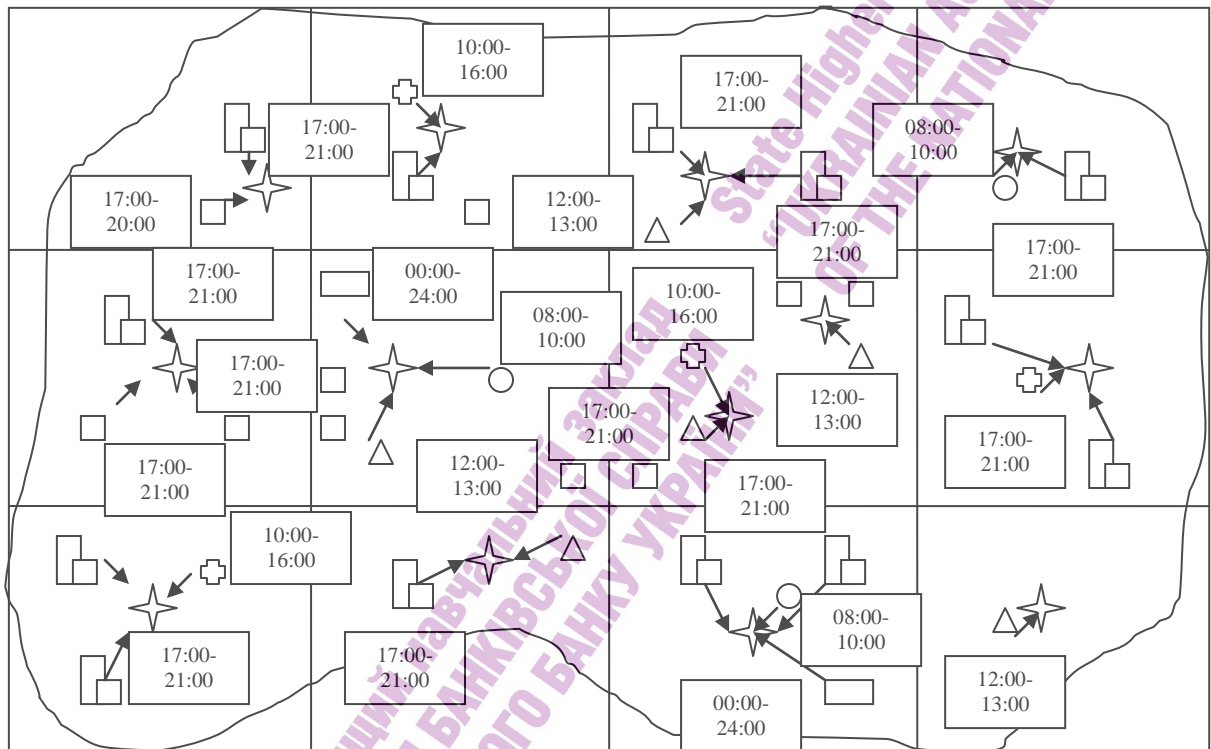
При використанні імітаційної моделі будуть зроблені деякі припущення:

- поведінка особи, що користується банкоматом, є раціональною, тому користувач буде орієнтуватися на близькість розміщення терміналу, а також на власний досвід роботи з банкоматом;



- моделювання проводиться для достатньо короткого періоду часу, що унеможливило кардинальну зміну середніх статистичних характеристик, що використовуються в моделі.

Складемо карту умовного міста, яку поділимо на квадрати, інтенсивність потоку клієнтів у яких буде різною, в залежності від кількості та масштабу об'єктів, розташованих у них. Зобразимо її на рисунку 2.4.



Умовні позначення:

- магазин
- ринок
- △ установа, яка бере участь у зарплатному проєкті
- ▭ вокзал
- ⊕ лікарня
- ▭ житловий масив
- ★ банкомат

Рисунок 2.4 – Карта умовного міста

Також позначимо на карті об'єкти, які впливають на інтенсивність використання банкоматів, потоки людей, наявні банкомати та місця

можливого встановлення нових банкоматів. Для кожного такого об'єкту позначимо час найбільшої активності людей щодо отримання готівки.

Використовуючи розроблену карту та запропоновану модель, можна визначити оптимальне розміщення банкоматної мережі. Даний процес може бути розбитий на декілька етапів.

На першому кроці на карті обираються місця найбільшої активності потенційних клієнтів, далі припускаються статистичні характеристики роботи даних банкоматів шляхом порівняння з аналогічними об'єктами.

За допомогою розробленої імітаційної моделі перевіряється доцільність встановлення банкоматів у запропонованому квадраті карти. При негативному результаті карта переглядається і знову проводиться моделювання.

Таким чином може бути визначено оптимальне розміщення мережі банкоматів на обраній території.



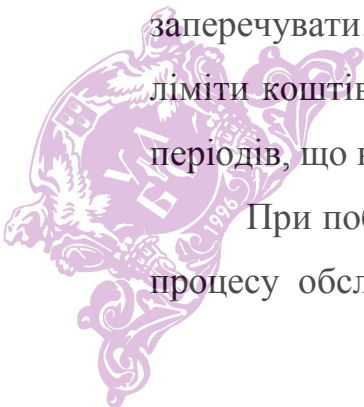
2.3 Аналіз запропонованих моделей

Як вже було зазначено, моделі визначення оптимальних сум та строків забезпечення банкоматів готівкою, які використовуються на сьогоднішній день банками у практичній діяльності, надають можливість врахування чинників, які мають досить велике значення для роботи банку. Так, необхідно розраховувати суму коштів, необхідну для забезпечення банкоматів таким чином, щоб використання грошей було найбільш ефективним. Мається на увазі, що період перебування грошей в банкоматі повинен бути таким, щоб кошти використовувалися повністю і перебували в обігу, інакше вони не принеситимуть користі. Але потрібно також зважати на той факт, що часте обслуговування банкоматів вимагає додаткових витрат, що теж не є бажаним. У практиці «Приватбанку» було вираховано, що оптимальний термін перебування грошей у банкоматі дорівнює приблизно трьом дням. Таким чином, дана методика при здійсненні розрахунку лімітів наявних коштів використовує нормативний коефіцієнт оборотності.

Що стосується моделі, яка базується на теоретико-ймовірнісному підході, то вона не встановлює рамок для проміжків часу між завантаженням банкоматів грошами, а надає можливість розрахувати ці терміни. Вона, однак, не виключає можливості зворотних розрахунків і дозволяє моделювати ситуацію таким чином, щоб час наближувався до бажаного.

На відміну від практичної методики, використовуваної банком, теоретико-ймовірнісна модель дає можливість розрахувати суму коштів, необхідну для забезпечення банкоматів з урахуванням усіх стохастичних чинників, вплив яких на будь-які економічні процеси неможливо заперечувати. У той самий час модель банку застосовує такий підхід, коли ліміти коштів для банкоматів розраховуються, виходячи тільки з попередніх періодів, що не завжди дає найкращий результат у поточному періоді.

При побудові теоретико-ймовірнісної моделі враховуються властивості процесу обслуговування, які є характерними для обслуговування клієнтів



банку за допомогою мережі банкоматів. Вони описані законами розподілу, що підтверджені науковими дослідженнями та перевірені на практиці.

Однак, слід відзначити, що дана модель є суто аналітичною і містить ряд припущень та асимптотичних розв'язків, тому більш доцільним є поєднання її із статистичним моделюванням, яке є по суті експериментом і дозволяє перевірити правильність аналітичної частини.

Таким чином, визначення оптимальних сум та строків завантаження банкоматів готівкою є досить актуальною задачею. Для реалізації цієї задачі найбільш доцільно проводити імітаційне моделювання. Для використання запропонованої моделі та перевірки її адекватності необхідно розробити автоматизований додаток, який буде проводити імітаційне моделювання ефективного обслуговування банкоматів, а у поєднанні з картою інтенсивності потоків клієнтів буде слугувати для визначення оптимального розміщення банкоматної мережі. У подальших розділах роботи буде висвітлено процес розробки автоматизованого додатку та проведення моделювання з його використанням.

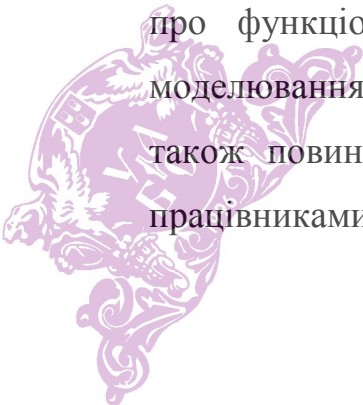


3 РОЗРОБКА АВТОМАТИЗОВАНОГО ДОДАТКУ МОДЕЛІ ЕФЕКТИВНОГО ОБСЛУГОВУВАННЯ БАНКОМАТІВ ТА ЇХ ОПТИМАЛЬНОГО РОЗМІЩЕННЯ

3.1 Інформаційне, організаційне і програмне забезпечення

На сьогоднішній день кожен банк користується апаратним і програмним забезпеченням, яке дозволяє не тільки швидко і якісно обробляти інформацію, але й зберігати її для подальшого використання. Це стосується і даних щодо роботи банкоматів, особливо, якщо банк досить великий і має розгалужену банкоматну мережу. У цьому випадку банк зберігає відомості про суми коштів, що завантажувалися у банкомати, про суми, що видавалися клієнтам, про періоди між завантаженнями банкоматів, про час надходження клієнтів, про тривалість обслуговування клієнтів тощо.

Уся ця інформація може бути корисною при проведенні імітаційного моделювання, яке потребує використання статистичних даних для розрахунку необхідних показників. При цьому досить важливо, щоб статистика була зібрана за тривалий проміжок часу і знаходилася у впорядкованому стані. Зазвичай інформацію зберігають у базі даних. Це потребує спеціальних знань і навиків від працівників, які обслуговують ці бази. Найчастіше питаннями, що стосуються банкоматів займаються співробітники інформаційних підрозділів банків, оскільки саме вони займаються питаннями, пов'язаними з банкоматами, або співробітники спеціальних відділів з обслуговування банкоматних мереж, якщо банк має дуже велику і розгалужену мережу. Проте обробкою і збереженням даних про функціонування та обслуговування банкоматної мережі, а також моделюванням і розрахунком необхідних показників роботи банкоматів також повинні займатися працівники аналітичного відділу у взаємодії з працівниками вищезазначених відділів.



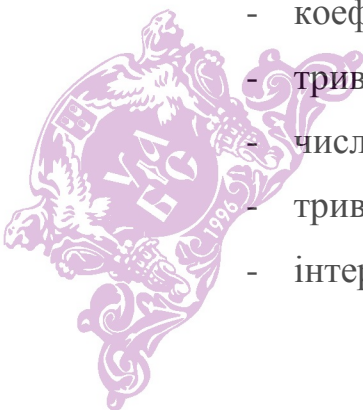
Рух інформації, яка стосується функціонування банкоматів, зображено на рисунку 3.1.



Рисунок 3.1 – Рух інформації щодо банкоматів

Для проведення імітаційного моделювання необхідна наступна інформація:

- початкова сума у банкоматі;
- критична сума у банкоматі;
- середня сума операції;
- коефіцієнт варіації суми операції;
- середня тривалість операції;
- коефіцієнт варіації тривалості операції;
- тривалість циклу моделювання у добах;
- число реалізацій циклу моделювання;
- тривалість операції завантаження банкомату;
- інтервал виконання запиту на завантаження.

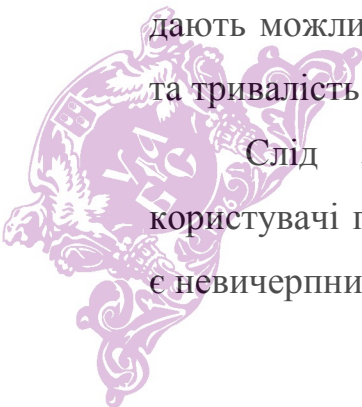


У результаті моделювання будуть отримані наступні показники:

- середня кількість клієнтів у день;
- середня кількість обслужених клієнтів у день;
- середня кількість необслужених клієнтів у день;
- середній період між завантаженнями банкомата;
- середній залишок коштів при інкасації;
- середня довжина черги;
- коефіцієнт використання банкомата;
- коефіцієнт простою через очікування клієнтів;
- коефіцієнт простою через вичерпання коштів;
- коефіцієнт простою через завантаження банкомату;
- тривалість моделювання у добах;
- абсолютна пропускна спроможність банкомату;
- відносна пропускна спроможність;
- щільність потоку клієнтів;
- інтенсивність обслуговування.

Отримавши вищезазначені показники, можна робити висновки про роботу мережі банкоматів і шляхи її вдосконалення. Для кращого розуміння доцільно проводити імітаційне моделювання декілька разів, змінюючи різні вхідні дані, для отримання результатів, що наближають обрані критерії роботи банкоматів до нормативів. До того ж, маючи карту міста, описану у другому розділі, і використовуючи результати імітаційного моделювання, можна визначити, де необхідно встановити додаткові банкомати чи зняти вже існуючі. Також, працівники відділу обслуговування банкоматної мережі отримують відомості про експлуатацію і обслуговування банкоматів, які дають можливість оцінити достатність сум, які завантажуються у банкомат, та тривалість їх використання.

Слід зазначити, що вищезазначений перелік показників, які користувачі програми, що проводить моделювання, отримують в результаті, є невичерпним, і за необхідністю працівники інформаційного та аналітичного

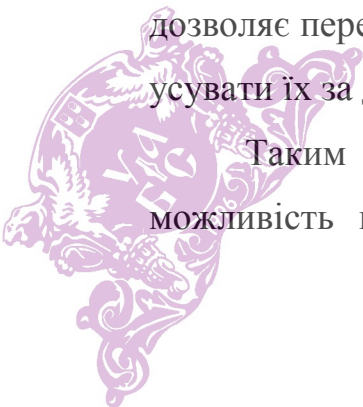


відділів можуть доопрацювати модель і додати інші показники. Для забезпечення такої можливості було здійснено пошук програмних засобів для реалізації моделі. При цьому було розглянуто програмне забезпечення, описане нижче.

На сьогодні існує безліч програмних засобів, які дозволяють вирішувати задачі, що містять складні математичні розрахунки, а також проводити математичне моделювання. Серед них безперечно найбільшою популярністю користуються спеціальні математичні пакети типу Mathematica, MatLab, Maple, Mathcad, Gauss, Reduce, Eureka та інші. Також не слід забувати й про традиційний MS Excel, який також є широко застосовуваним і має великий набір функцій, у тому числі, й математичних.

Окрім цього, досить поширеним є використання при моделюванні різноманітних мов програмування і прикладних програм, що надають середовище їх використання, оскільки розроблене при цьому програмне забезпечення має ряд переваг. Одним із таких середовищ є Delphi 7, яке також надає широкі можливості для вирішення поставлених задач. Так, на відміну від вищеописаних прикладних пакетів, воно надає можливість точно описати алгоритм, наведений у одній з моделей, причому зробити це на дуже високому рівні представлення. Отже, в результаті програмування одержуємо цілісний програмний продукт з широкими можливостями і зручним інтерфейсом. Розроблена у середовищі Delphi 7 програма є самостійним продуктом і не вимагає встановлення додаткових прикладних пакетів для її використання. Що стосується мови програмування, представленої у Delphi 7, вона є досить простою, надає можливість описувати складні алгоритми, використовувати об'єктно-орієнтований підхід тощо. Окрім того, Delphi 7 дозволяє перевіряти окремі складові алгоритму, швидко знаходити помилки і усувати їх за допомогою трасування.

Таким чином, порівнявши декілька програмних засобів, які дають можливість проводити складні математичні розрахунки, можна зробити



наступні висновки про їх придатність для вирішення задачі, поставленої у даній роботі.

MS Excel в цілому є дуже зручним засобом розв'язання деяких математичних задач, але у нашому випадку він значно програє перед іншими програмними пакетами, оскільки не призначений для реалізації моделей подібної складності.

Спеціалізовані математичні пакети надають широкий спектр вбудованих математичних функцій, проте вони не зручні при описанні складних циклічних алгоритмів.

Delphi 7 у цьому відношенні є найбільш зручним і має можливості для проведенні всіх необхідних розрахунків, а тому саме цей програмний засіб було обрано для створення практичної частини даної роботи.

3.2 Проведення експериментування з моделлю на тестовому прикладі

В ході роботи було розроблено програмний додаток для імітаційного моделювання роботи банкомату як системи масового обслуговування.

У результаті роботи даної програми користувач отримує статистичні показники, які можуть бути використані у практичній діяльності банку щодо організації ефективного функціонування мережі банкоматів.

Програму було розроблено у середовищі Delphi 7. Додаток містить дві форми, одна з яких є головною і відкривається на початку роботи користувача з програмою. Вона представлена на рисунку 3.2.

На даній формі знаходиться область введення вхідних даних моделі, де користувач повинен задати усі початкові параметри моделі.

У нижній частині головного вікна знаходиться робоча область, представлена полем відліку часу, полем відображення згенерованої програмою кількості клієнтів, полем відображення маршрутів руху клієнтів та трьома функціональними кнопками.

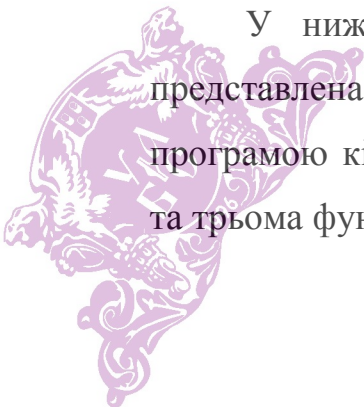


Рисунок 3.2 – Головна форма програми.

Для позначення подій щодо руху клієнтів введено деякі позначення:

- TO QUEUE – клієнт становиться у чергу;
- DIRECTLY to terminal – черга пуста, і клієнт одразу надходить на обслуговування;
- FROM QUEUE to terminal – клієнт надходить з черги на обслуговування;
- OUT from terminal – клієнт закінчує обслуговування і залишає банкомат;
- QUEUE IS TOO BIG, GO OUT – черга є надмірно великою, тому клієнт уходить, не становлячись до черги;

- RECHARGING, GO OUT – банкомат перезавантажується, тому клієнт уходить, не становлячись в чергу;

- RELEASE QUEUE – черга розходиться через обслуговування банкомату.

При цьому структура записів у вікні маршрутів є такою:
/номер клієнта/ /№ доби циклу/ /година доби/ /подія/ /тривалість обслуговування/ /сума операції/.

Дана структура може містити не всі компоненти, в залежності від події.

До того ж, у робочій області знаходиться поле, у якому відображується номер реалізації моделі.

Усі кнопки робочої області головної форми дублюються відповідними пунктами меню.

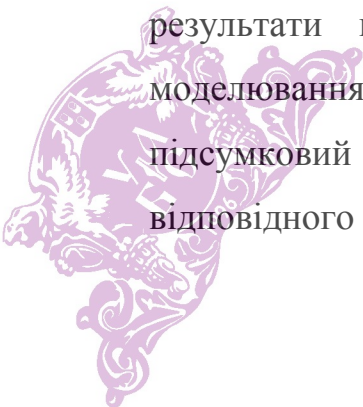
Якщо у користувача виникають складнощі у роботі з програмою, він має можливість з головного вікна програми викликати довідку, скориставшись відповідним пунктом меню, у якій детально описані функції програми та призначення її компонентів. Довідкова система має наступні розділи:

- про програму;
- введення вхідних даних;
- обробка даних;
- результати моделювання.

При роботі з довідкою користувач може користуватися змістом та предметним вказівником або ж викликати пошук за словами.

Вигляд одного з вікон довідки представлено на рисунку 3.3.

Окрім головної форми, додаток має окрему форму, на якій виводяться результати моделювання. Отримані показники після кожного циклу моделювання виводяться у окремому стовпчику таблиці. Кожний підсумковий показник представлено у окремому текстовому полі навпроти відповідного показника в таблиці.



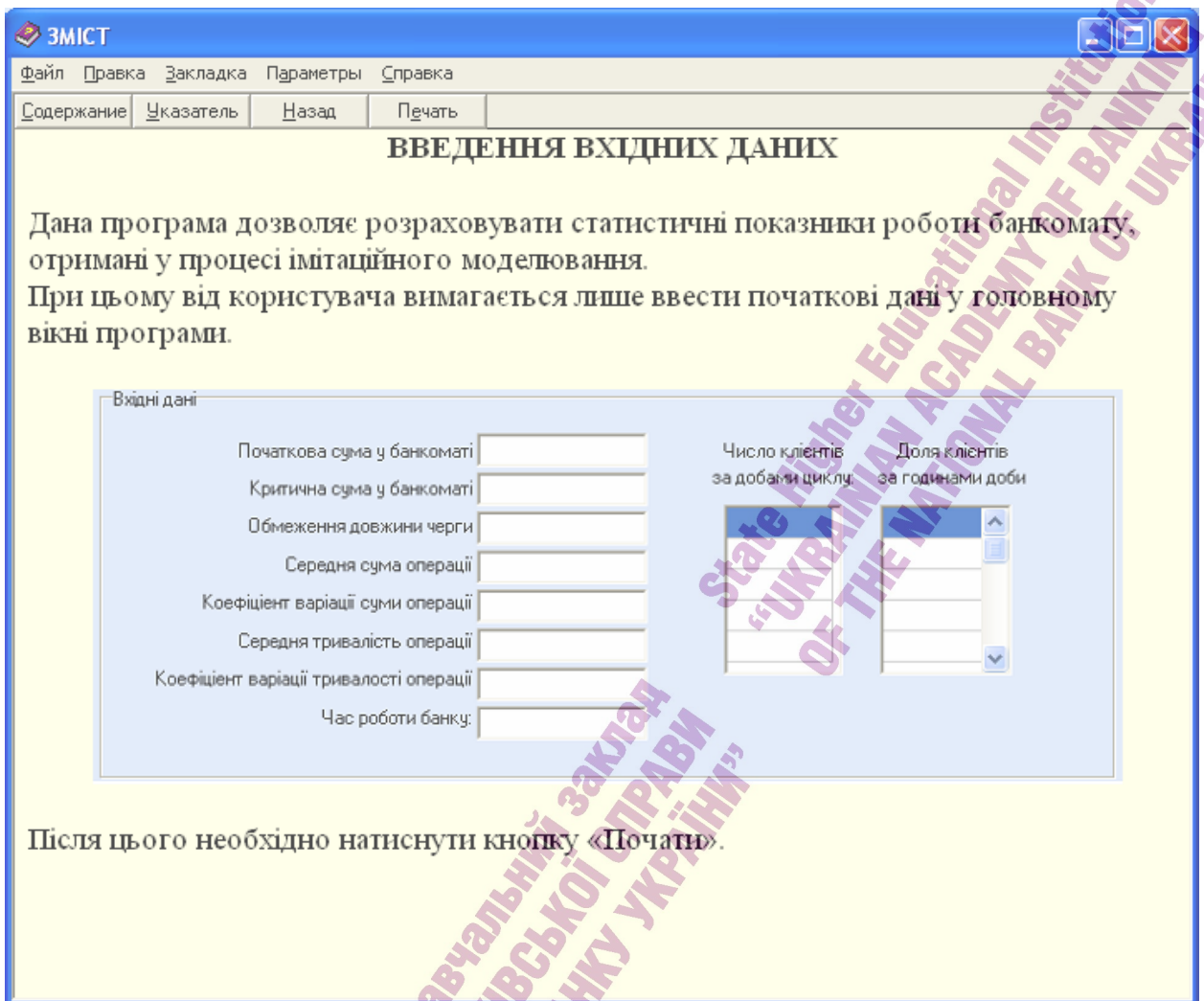
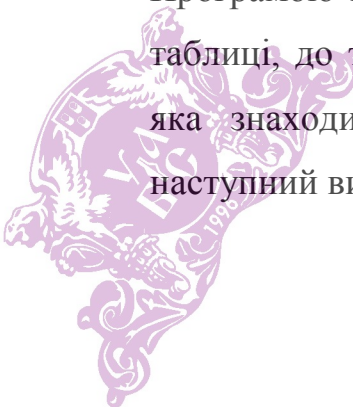


Рисунок 3.3 – Вікно довідкової системи програмного додатку

Для зручності сприйняття стовпчики таблиці нумеруються відповідно до номеру реалізації моделювання.

На формі результатів знаходиться кнопка «Сховати», яка дозволяє сховати дане вікно без втрати даних, виведених на ній, тому викликати і закрити результатне вікно можна безліч разів, поки не буде проведено наступне моделювання або не буде закрито головне вікно програми. Програмою також передбачено збереження результатних даних, виведених у таблиці, до текстового файлу за допомогою кнопки «Зберегти результати», яка знаходиться поруч із кнопкою «Сховати». Вікно результатів має наступний вигляд.



Результати моделювання

Номер реалізації:

Середня кількість клієнтів у день, чол.:

Середня кількість обслужених клієнтів, чол.:

Кількість необслужених клієнтів, чол.:

з них: через недостатність грошей:

через необхідність
перезавантаження банкомату:

під час перезавантаження банкомату:

через обмеження черги:

Середня кількість клієнтів в черзі, чол.:

Середній час перебування в черзі, хв.:

Середній час обслуговування, хв.:

Середній період завантаження банкомату, год.:

Середній залишок при інкасації, грн.

Абсолютна пропускна спроможність, чол.:

Відносна пропускна спроможність:

Щільність вхідного потоку чол./год.:

Інтенсивність обслуговування чол./год.:

Коефіцієнт використання банкомату:

Коефіцієнт простою через очікування клієнтів:

Коефіцієнт простою через нестачу коштів:

Коефіцієнт простою через завантаження:

Сховати Зберегти результати

Рисунок 3.4 – Вікно результатів моделювання

Отже, на початку роботи з програмою необхідно заповнити усі поля вхідної інформації, після чого натиснути кнопку «Почати» або скористатися відповідним підпунктом у пункті меню «Моделювання». Після закінчення обробки даних користувач має можливість переглянути результати моделювання. Для цього необхідно натиснути кнопку «Показати результати» або скористатися підпунктом «Результати» у пункті меню «Моделювання». Користувач побачить результатну форму і зможе закрити її за допомогою кнопки «Сховати». При цьому дані, відображені у полях даної форми не будуть втрачені, і їх знов можна буде побачити, викликавши форму результатів.

Для закінчення роботи з програмою необхідно натиснути кнопку «Вихід» або скористатися відповідним пунктом меню.

Для перевірки роботи програми та адекватності моделі розглянемо тестовий приклад.

Запустимо додаток і введемо початкові дані. Дані про роботу банкомату наведено у таблиці 3.1.

Таблиця 3.1 – Вхідні дані щодо роботи банкомату.

№ п/п	Показник	Значення
1	Початкова сума у банкоматі, грн.	50000
2	Критична сума у банкоматі, грн.	1000
3	Обмеження довжини черги, чол.	10
4	Середня сума операції, грн.	100
5	Коефіцієнт варіації суми операції	1,358
6	Середня тривалість операції, хв.	1,587
7	Коефіцієнт варіації тривалості операції	0,988
8	Час роботи банку, год.	120
9	Кількість реалізацій	100

Дані про надходження клієнтів на обслуговування наведено у таблицях 3.2 і 3.3.

Таблиця 3.2 – Кількість клієнтів за добами циклу

№ доби п/п	Число клієнтів, чол..
1	85,5
2	74,25
3	114,75
4	131,4
5	255,75



Таблиця 3.3 – Частка клієнтів за годинами доби

Година доби	Частка клієнтів за годинами доби	Година доби	Частка клієнтів за годинами доби	Година доби	Частка клієнтів за годинами доби
1	0,005	9	0,093	17	0,097
2	0,005	10	0,053	18	0,061
3	0,005	11	0,055	19	0,077
4	0,005	12	0,055	20	0,027
5	0,005	13	0,019	21	0,014
6	0,006	14	0,010	22	0,011
7	0,016	15	0,058	23	0,007
8	0,027	16	0,079	24	0,010

Вигляд головної форми програми в процесі обробки даних представлено на рисунку 3.5.

Робота програми при 100 реалізаціях триває приблизно 20 хвилин. Після закінчення моделювання виводиться відповідне повідомлення. Результатну форму із отриманими показниками представлено на рисунку 3.6. Результати моделювання занесемо в таблицю 3.4.

Таблиця 3.4 – Результати імітаційного моделювання

№ п/п	Назва показнику	Значення
1	Середня кількість клієнтів у день, чол.	123,37
2	Середня кількість обслужених клієнтів, чол.	98,92
3	Середня кількість необслужених клієнтів, чол.	24,42
4	з них: через недостатність коштів	0,22
5	через необхідність перезавантаження банкомату	1,37
6	під час перезавантаження банкомату	2,73
7	через обмеження черги	19,61
8	Середня кількість клієнтів в черзі, чол.	2,93
9	Середній час перебування в черзі, хв.	1,98
10	Середній час обслуговування, хв.	1,99
11	Середній період завантаження банкомату, год.	63,6

Продовження таблиці 3.4

№ п/п	Назва показнику	Значення
12	Середній залишок при інкасації, грн.	434,1
13	Абсолютна пропускна спроможність, чол.	4,11
14	Відносна пропускна спроможність, чол./год.	0,8
15	Щільність вхідного потоку, чол./год.	5,14
16	Інтенсивність обслуговування, чол./год.	4,12
17	Коефіцієнт використання банкомату	0,246
18	Коефіцієнт простою через очікування клієнтів	0,696
19	Коефіцієнт простою через нестачу коштів	0
20	Коефіцієнт простою через завантаження	0,058

Імітаційна модель системи банківського обслуговування

Моделювання Довідка Вихід

Вхідні дані:

Початкова сума у банкоматі, грн.	50000	Число клієнтів за добами циклу:	85,5	Доля клієнтів за годинами доби	0,005
Критична сума у банкоматі, грн.	1000		74,25		0,005
Обмеження довжини черги, чол.	10		114,75		0,005
Середня сума операції, грн.	100		131,4		0,005
Коефіцієнт варіації суми операції	1,358		255,75		0,005
Середня тривалість операції, хв.	1,587				
Коефіцієнт варіації тривалості операції	0,988				
Час роботи банку, год.	120				
Кількість реалізацій	100				

Відлік часу: 61

Кількість клієнтів: 637

Номер реалізації: 7

Маршрути клієнтів банку:

```
# 187 day 3 hour 15 TO QUEUE
# 188 OUT from terminal
# 187 day 3 hour 15 FROM QUEUE to terminal serv time 7 amount 227
# 187 OUT from terminal
# 189 day 3 hour 15 DIRECTLY to terminal serv time 4 amount321
# 189 OUT from terminal
# 191 day 3 hour 16 DIRECTLY to terminal serv time 4 amount85
# 190 day 3 hour 16 TO QUEUE
# 191 OUT from terminal
# 190 day 3 hour 16 FROM QUEUE to terminal serv time 2 amount 92
# 193 day 3 hour 16 TO QUEUE
# 192 day 3 hour 16 TO QUEUE
# 190 OUT from terminal
# 193 day 3 hour 16 FROM QUEUE to terminal serv time 1 amount 355
```

Почати

Показати результати

Вихід

Рисунок 3.5 – Головна форма програми в процесі обробки даних

Результати моделювання		Номер реалізації:										
		1	2	3	4	5	6	7	8	9	10	11
Середня кількість клієнтів у день, чол.:	120,94	97,2	131,8	120,6	127,8	125,8	118	124	116	122,4	118,2	117,2
Середня кількість обслужених клієнтів, чол.:	97,94	73	106,2	99,6	103,8	98,8	97,6	99,2	94	100	101	100
Кількість необслужених клієнтів, чол.:	23	24,2	25,6	21	24	27	20,4	24,8	22	22,4	17,2	20,2
з них: через недостатність грошей:	0,41	0	0,8	1	0,8	1,2	0	0	0	0	0,8	0
через необхідність перезавантаження банкомату:	1,63	1,8	1,8	1	1,8	1,8	1,8	1,8	1,4	1,8	1,6	1,1
під час перезавантаження банкомату:	2,67	0,4	4,6	3,8	2,6	4,2	1,6	1,8	3,2	2,6	1	3,1
через обмеження черги:	18	22	18,6	14	18,4	20	17	19,8	17,4	18	13,8	19,1
Середня кількість клієнтів в черзі, чол.:	2,92	2,35	3,23	2,87	3,02	2,99	2,96	2,9	2,74	2,82	3	3,1
Середній час перебування в черзі, хв.:	1,34	1,242	1,244	1,595	1,27	1,313	1,477	1,346	1,418	1,444	1,396	1,418
Середній час обслуговування, хв.:	1,97	1,947	1,976	1,987	2,014	1,963	1,995	1,967	1,983	1,995	1,95	1,976
Середній період завантаження банкомату, год.:	65,45	120	60	60	60	60	60	60	60	60	60	60
Середній залишок при інкасації, грн.:	240,81	791	178	407	123,5	186,5	252,5	0	339	0	269,5	100
Абсолютна пропускна спроможність, чол.:	4,07	3,033	4,417	4,142	4,317	4,108	4,058	4,125	3,908	4,158	4,2	4,1
Відносна пропускна спроможність:	0,80	0,749	0,804	0,824	0,811	0,784	0,825	0,798	0,809	0,815	0,853	0,804
Щільність вхідного потоку чол./год.:	5,04	4,05	5,49	5,03	5,33	5,24	4,92	5,17	4,83	5,1	4,92	5,1
Інтенсивність обслуговування чол./год.:	4,08	3,04	4,42	4,15	4,33	4,12	4,07	4,13	3,92	4,17	4,21	4,1
Коефіцієнт використання банкомату:	0,027											
Коефіцієнт простою через очікування клієнтів:	0,696											
Коефіцієнт простою через нестачу коштів:	0											
Коефіцієнт простою через завантаження:	0,058											

Рисунок 3.6 – Результатна форма

Таким чином, за допомогою спеціально розробленого програмного додатку було здійснено імітаційне моделювання експлуатації і обслуговування банкомату. Кількість реалізацій моделі дорівнювала 100, що є достатнім для забезпечення адекватності результатних показників. У процесі роботи програми було окремо розраховано показники за кожною реалізацією, а після цього – підсумкові показники, які представляють собою середні значення результатних показників, а також спеціальні коефіцієнти.

За отриманими показниками можна робити висновки про навантаження на банкомат, про рух коштів у банкоматі та інші, що можуть бути використані фахівцями при прийнятті управлінських рішень стосовно ефективного обслуговування мережі банкоматів банку.



3.3 Оцінка ефективності імітаційної моделі і аналіз результатів

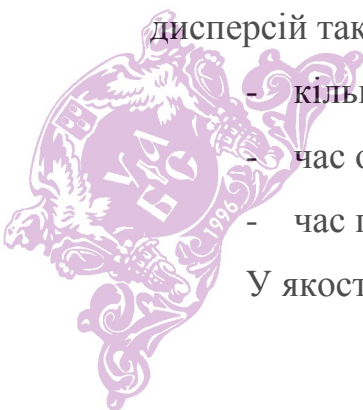
Як показали попередні дослідження, використання імітаційної моделі у практичній діяльності банку може бути достатньо корисним. Проте слід зазначити, що перш ніж застосовувати будь-яку модель, необхідно перевірити її адекватність. В ході роботи було перевірено адекватність обраної імітаційної моделі методом оцінки однорідності дисперсій [27]. Даний метод полягає у перевірці гіпотези щодо належності N вибірових дисперсій результатних показників експериментальних досліджень до однієї генеральної сукупності. Для перевірки цієї гіпотези використовується критерій Кохрена. Гіпотезу про однорідність вибірових дисперсій за критерієм Кохрена перевіряють за такою схемою:

- проводять декілька експериментів, послідовно змінюючи на невелику величину значення обраних факторів;
- при цьому для кожного експерименту обраховуються дисперсії обраних результатних показників;
- серед ряду дисперсій кожного показника знаходять найбільшу;
- обчислюють відношення найбільшої дисперсії до суми усіх дисперсій G ;
- визначають число ступенів вільності f_1 і f_2 ;
- обирають рівень значущості q і за даними f_1 , f_2 та q у спеціальній таблиці знаходять величину критичного відношення $G_{кр}$;
- порівнюють G і $G_{кр}$: якщо $G \leq G_{кр}$, то гіпотеза про однорідність дисперсій приймається, якщо $G > G_{кр}$, гіпотеза відкидається.

За вищеописаною схемою було проведено перевірку однорідності дисперсій таких результатних показників моделі, як

- кількість обслугованих клієнтів,
- час обслуговування,
- час перебування в черзі.

У якості змінних вхідних параметрів експериментування було обрано



- початкову суму у банкоматі,
- середню суму;
- середню тривалість операції.

Було проведено вісім експериментів, у кожному з яких послідовно змінювалися вхідні параметри у визначеному діапазоні.

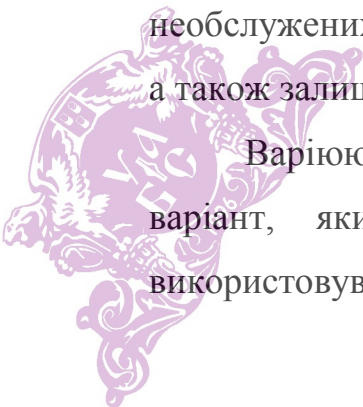
У результаті проведення перевірки за критерієм Кохрена було виявлено, що відношення найбільшої дисперсії до суми усіх дисперсій кожного результатного показника менше критичного значення, взятого з таблиці при рівні значущості 0,05 ($G_{кр}=0,5157$, $G_1=0,1549$, $G_2=0,1793$, $G_3=0,1969$). А отже, гіпотеза про однорідність дисперсій підтвердилася, а тому обрану імітаційну модель можна вважати адекватною.

Під час проведення імітаційного моделювання на тестових прикладах використовувалися різні вхідні дані з метою визначити їх вплив на результати моделювання. Було отримано результатні показники, перелік яких зазначався вище. При цьому зміна таких вхідних даних, як первинна сума грошей у банкоматі або критична сума у банкоматі суттєво впливала на результати.

Так, при збільшенні початкової суми у банкоматі за незмінної критичної суми результатні показники покращилися: зросло число обслужених клієнтів, зменшилася кількість клієнтів, що були не обслужені через нестачу коштів та перезавантаження банкомату, збільшився період між перезавантаженнями банкомату, зменшились також значення коефіцієнтів простою банкомату через нестачу коштів і через перезавантаження.

Зменшення критичної суми у банкоматі за незмінної початкової суми головним чином впливало на такі показники, як кількість клієнтів, необслужених через нестачу коштів, період між завантаженнями банкомату, а також залишок коштів при інкасації.

Варіюючи, таким чином, вхідні дані, можна знайти такий оптимальний варіант, який найбільш задовольнить політику банку, який буде використовувати дану модель. Так, деякі банки прагнуть підвищити свій



імідж і покращити якість обслуговування, в тому числі і у мережі банкоматів. Тому для таких банків кращим шляхом буде збільшувати витрати на забезпечення банкоматів готівкою, оскільки прийдеться підвищити обсяги початкових сум у банкоматі, а до того ж доведеться посилити фінансування обслуговування банкоматів, адже критичні суми у банкоматах також не можна буде робити занадто малими. Проте, така політика властива невеликим, новим банкам, а також банкам з невеликою мережею банкоматів. Що стосується інших, зазвичай вони прагнуть скоротити обсяги коштів, які витрачаються на забезпечення роботи банкоматної мережі. Це не означає, що їх не цікавить якість обслуговування та заохочення клієнтів, але масштаби їх мереж банкоматів занадто великі для нераціонального використання грошових ресурсів. В якості прикладу подібних банків можна навести «Правекс-банк». Як свідчать літературні джерела [39], цей банк планує підвищити доходність банкоматної мережі шляхом зменшення альтернативних витрат, що виникають при використанні готівкових коштів на підкріплення банкоматів. Резерв збільшення доходності банкоматів керівництво банку оцінює у розмірі 300 грн. у місяць на кожний банкомат, а враховуючи масштаби мережі банкоматів, загалом може бути вивільнено біля 3 млн. готівкових коштів.

Таким чином, використовуючи імітаційну модель ефективного обслуговування і карту інтенсивності потоків клієнтів на визначеній території, можна визначити оптимальні суми і строки завантаження банкоматів готівкою, що значно скоротить витрати банку на обслуговування банкоматної мережі. До того ж, за допомогою вищезазначених інструментів можна не тільки оцінити параметри функціонування існуючих банкоматів, але й спрогнозувати ситуацію при встановленні нових терміналів.



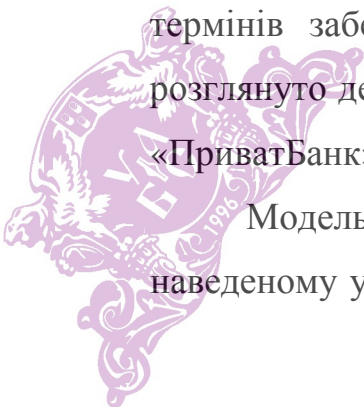
ВИСНОВКИ

В ході роботи було детально вивчено проблему ефективної організації роботи банкоматів і стан її вирішення банківськими установами, зокрема банком «Надра» та «ПриватБанком». Під час дослідження роботи систем банкоматів цих банків було виявлено, що їх функціонування не є досконалим. Так, до проблем в роботі банкоматів можна віднести такі, як прості банкоматів через нестачу коштів, великі залишки коштів у банкоматах під час інкасації, великі черги тощо. Для розв'язання цих проблем може використовуватися ряд заходів від моніторингу мережі банкоматів до розробки спеціальних методик. Під час ознайомлення з роботою вищезазначених банків було виявлено, що обслуговування мережі банкоматів банку «Надра» здійснюється шляхом постійного моніторингу банкоматної мережі і визначення сум, необхідних для підкріплення банкоматів ведеться, виходячи з попередніх періодів і досвіду працівників, які обслуговують банкомати. «ПриватБанк» використовує спеціальну методику розрахунку сум, необхідних для забезпечення роботи банкоматів, проте дана методика також базується лише на попередніх періодах, не враховуючи стохастичність процесу обслуговування.

В процесі пошуку кращої моделі для розв'язання задачі, поставленої у роботі, було досліджено ряд літературних джерел, проте слід зазначити, що у літературі обговорення цієї теми зустрічається досить рідко. Частіше банки використовують свої власні методики, які у більшості випадків є банківською таємницею і не публікуються, а до того ж не завжди є оптимальними.

В ході дослідження існуючих методик визначення оптимальних сум і термінів забезпечення банкоматів грошовими купюрами, було детально розглянуто декілька моделей, одна з яких використовується в роботі ЗАТ КБ «ПриватБанк», а інші на даний момент є більш теоретичними розробками.

Модель «ПриватБанку» є досить простою, але у її викладенні, наведеному у нормативному документі банку, є деякі недоліки і протиріччя,



які призводять до викривлення результатів розрахунків, що проводяться за даною методикою. Альтернативою для даної моделі є модель Васіна Н.С., яка використовує теоретико-ймовірнісний підхід до визначення оптимальних термінів і сум забезпечення банкоматів готівкою, хоча і вона не позбавлена деяких недоліків. Третя модель є поєднанням теоретико-ймовірнісної моделі і статистичного моделювання, що робить її більш досконалою, оскільки дозволяє провести своєрідний експеримент і отримати усереднені показники роботи банкоматної мережі.

Тому після детального вивчення даних моделей для реалізації було обрано саме останню модель, і за нею було розроблено повнофункціональний програмний додаток. Створена програма дає користувачу можливість розрахувати усі необхідні показники, лише ввівши початкові умови. Вона має дружній інтерфейс, нескладна у користуванні, а до того ж має довідкову систему для починаючих користувачів. Перевагою автоматизованого додатку є і те, що він дозволяє зберегти результатні дані у окремому файлі, якщо це необхідно.

Саме ця можливість програми була використана при перевірці імітаційної моделі на адекватність. Для цієї перевірки було обрано метод визначення однорідності дисперсій за критерієм Кохрена, у результаті чого було виявлено, що модель є адекватною.

Перевагою даної моделі є те, що вона не тільки враховує стохастичний характер процесу обслуговування клієнтів за допомогою мережі банкоматів, на відміну від моделі «ПриватБанку», яка у своїх розрахунках базується лише на даних попереднього періоду, але й представляє собою комп'ютерний експеримент, тобто дозволяє за допомогою автоматизованого додатку відтворити реальний процес обслуговування клієнтів банку у мережі банкоматів і отримати реалістичні результатні показники.

Також в ході роботи було розроблено умовну карту міста, на якій було позначено розташування клієнтів банку, їх активність та напрямки руху. Використання даної карти у поєднанні з імітаційним моделюванням дає



можливість оцінити ефективність функціонування наявних банкоматів, а також спрогнозувати ефективність роботи банкоматів, що планується встановити у певному районі міста. До того ж, саме за допомогою вищезазначеного інструментарію можна визначити оптимальне місце для розташування нових банкоматів.

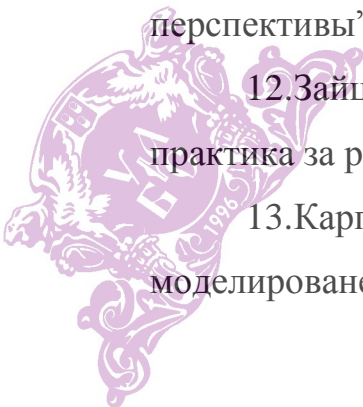
Що стосується ефективності обраної моделі, слід зазначити, що для різних банків критерії ефективності є різними, оскільки це залежить від їх місця на ринку банківських послуг і від політики керівництва. Проте, дана модель призначена для більш великих банків, оскільки саме в них виникають труднощі із забезпеченням ефективної експлуатації і обслуговування їх мереж банкоматів. Тому головними критеріями ефективності були суми залишків у банкоматах та періоди між їх завантаженнями. Отже, суми залишків у банкоматах повинні бути мінімальними, а оптимальні періоди між завантаженнями банкоматів визначаються у межах 3-4 днів, як свідчить банківська практика. При цьому необхідно також враховувати і такі показники, як коефіцієнти простою банкоматів внаслідок різних причин. Вони повинні бути мінімальними.

Таким чином, для підвищення ефективності роботи мережі банкоматів необхідно використовувати імітаційне моделювання у поєднанні з теорією масового обслуговування та картою інтенсивності потоку клієнтів на визначеній території. Поєднання саме цих інструментів дає можливість зменшити витрати на обслуговування мереж банкоматів і ефективно організувати розташування банкоматів всередині мережі.



ПЕРЕЛІК ПОСИЛАНЬ

1. Бакнелл Джулиан М. Фундаментальные алгоритмы и структуры данных в Delphi : Пер. с англ. / Джулиан М. Бакнелл. – СПб: ООО «Диа Софт ЮП», 2003 – 560с.
2. Бюллетень НБУ. – 2007. – №4. – С.24-25.
3. Васин Н.С. «Статистическое моделирование функционирования банкоматных систем» // Финансы и кредит. – 2006. – №4. – С.56-61.
4. Васин Н.С. «Теоретико-вероятностный анализ и прогнозирование резерва наличности для обеспечения банкоматных операций» // Финансы и кредит. – 2005. – №25. – С.68-70.
5. Васин Н.С. «Теоретико-вероятностный анализ и прогнозирование сроков подкрепления банкоматов наличностью» // Финансы и кредит. – 2005. – №27. – С.55-57.
6. Вентцель Е.С., Овчаров Л.А. Теория вероятностей. – М.: Наука, 1969. – 368с.
7. Вересюк А. «Новая роль банкоматов» //Банковская практика за рубежом (рус.). - 2003.-№ 6.-С.60-62.
8. Вітлінський В.В. Моделювання економіки: Навч. посібник. – К.: КНЕУ, 2003. – 408 с.
9. Гмурман В.Е. Теория вероятностей и математическая статистика. Учеб. пособие для вузов. Изд. 6-е, стер. – М.: Высш. шк., 1998. – 479 с.
10. Дубина А.Г., Орлова С.С. Excel для экономистов и менеджеров. – СПб.: Питер, 2004. – 295с.
11. Євгенія Лакосник „Нейтральне банкоматы: возможности и перспективы” //Банковская практика за рубежом (рус.). - 2005.- № 8.-С.54- 58.
12. Зайцев О. «АТМ: функции новые – роль та же» //Банковская практика за рубежом (рус.). - 2004.-№ 10.- С.81-86.
13. Карпов Ю. Имитационное моделирование систем. Введение в моделирование с AnyLogic5 – СПб.: БХВ-Петербург, 2005 – 400с.



- 14.Кобелев Н.Б. Основы имитационного моделирования сложных систем. – М: «Дело», 2003. – 336с.
- 15.Компьютеризация банковской деятельности під редакцією Титоренко Г.А.- М.: Финстатинформ, 1997.- 304 с.
- 16.Костіна Н.І., Антонов В.М., Ганах Н.І. Банки: сучасні інформаційні технології: Навчальний посібник - К.: Ірпінь, 2001.- 359 с.
- 17.Кремер Н.Ш. Теория вероятностей и математическая статистика: Учебник. – М.: «Юнити-Дана», 2007. – 573с.
- 18.Міщенко В., Набок Р., Шитко О. «Проблеми організації емісії та обігу готівки» // Вісник НБУ. – 2007. –№1. – С.17-23.
- 19.Одарюк А. «Пожиратели купюр» //Бизнес. – 2004. – №6. – С.46-47.
- 20.Одарюк А. «Убратъ кассира» //Бизнес. – 2004. – №40. – С.52-54.
- 21.Олег Зайцев „АТМ: функции новые – роль та же” //Банковская практика за рубежом (рус.). - 2004.-№ 10.- С.81-86.
- 22.Олексій Вересюк „Новая роль банкоматов” //Банковская практика за рубежом (рус.). - 2003.-№ 6.-С.60-62.
- 23.Очков В.Ф. Mathcad 7 Pro для студентов и инженеров. – М.: КомпьютерПресс, 1998. – 384 с.
- 24.Плис А.И., Сливина Н.А. Mathcad. Математический практикум для инженеров и экономистов: Учеб. пособие. – 2-е изд., перераб. и доп. – М.: Финансы и статистика, 2003. – 656 с.
- 25.Романенко Л.Ф. „Стан та перспективи розвитку сучасного ринку пластикових карток в Україні” //Економіка і прогнозування. - 2001.-№4.- С.95-102.
- 26.Стивенс Р. Delphi. Готовые алгоритмы : Пер. с англ. – М.:ДМК Пресс, 2001. – 384с.
- 27.Ситник В.Ф., Орленко Н.С. Імітаційне моделювання: Навчальний посібник. – К.:КНЕУ, 1998. – 232с.



28.Тейксейра С., Пачеко К. Delphi 5. Руководство разработчика: В 2-х т. Т.1.: Основные методы и технологии: Учебное пособие:Пер. с англ. – М.: Вильямс, 2000.- 832 с.

29.Томашевский В.Н. Имитационное моделирование систем и процессов. – К.: ІСДО, «ВІПОЛ», 1994. – 124с.

30.Томашевський В.М. Моделювання систем. – К.: Видавнича група ВНУ, 2005. – 352с.

31.Федосеев В.В. Экономико-математические методы и прикладные модели: Учеб. пособие для вузов. – М.: ЮНИТИ, 1999. – 391 с.

32.Федоткин М.А. Основы прикладной теории вероятностей и статистики. Учебник. – М.: Высшая школа, 2006. – 368с.

33.Феллер В. Введение в теорию вероятностей и ее приложения. В 2 т. Т.1. – М.: Мир, 1964. – 498с.

34.Феллер В. Введение в теорию вероятностей и ее приложения. В 2 т. Т.2. – М.: Мир, 1964. – 752с.

35.Харченко В. «Підсумки діяльності банків України на ринку платіжних карток у 2006 році» // Вісник НБУ. – 2007. –№4. – С.2-9.

36.Хинчин А.Я. Работы по математической теории массового обслуживания – М.: Физматгиз, 1963. – 236с.

37.Рут Силверман «Практикум по моделированию на GPSS/H для работ по профилирующей дисциплине компьютерные науки» // <http://simulation.org.ua/methods.php>

38.Соловьев А.В. «Математическое моделирование на примере системы массового обслуживания типа G/G/m/n по Кенделу» // <http://kit-2001.narod.ru>

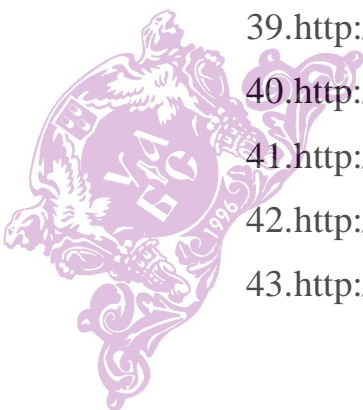
39.<http://imperia.net.ua>

40.<http://modelsmo.narod.ru/old/index.html>

41.<http://ntcnvg.ru/lekcii>

42.<http://uk.wikipedia.org/wiki>

43.<http://www.banker.com.ua>



44. <http://www.bitec.spb.ru>

45. <http://www.exponenta.ru>

46. <http://www.nkzu.edu>

47. <http://yevgeny.nm.ru/institut.html/model.html>

State Higher Educational Institution
“UKRAINIAN ACADEMY OF BANKING
OF THE NATIONAL BANK OF UKRAINE”

Державний вищий навчальний заклад
“УКРАЇНЬКА АКАДЕМІЯ БАНКІВСЬКОЇ СПРАВИ
НАЦІОНАЛЬНОГО БАНКУ УКРАЇНИ”



ДОДАТКИ

**Державний вищий навчальний заклад
“УКРАЇНЬКА АКАДЕМІЯ БАНКІВСЬКОЇ СПРАВИ
НАЦІОНАЛЬНОГО БАНКУ УКРАЇНИ”**

**State Higher Educational Institution
“UKRAINIAN ACADEMY OF BANKING
OF THE NATIONAL BANK OF UKRAINE”**



Додаток А

Лістинг програмного додатку

```

unit Unit1;
interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
  Controls, Forms,
  Dialogs, ExtCtrls, StdCtrls, Math, Grids, Unit2, Menus;
type
  PClient=^TClient;
  TWorld=class
  private
    clients:array of PClient;
    clients_count:integer;
    accumulate_time:int64;
    ro:real;
    alpha:real;
    eta:real;
    teta:real;
    procedure CreateClients;
    procedure StartTimers;
  public
    served_clients:integer;
    constructor Create;
    procedure Start;
    function LogNormal:real;
    function Gamma:real;
  end;
  TClient=class
  private
    id:integer;
    me:PClient;
    term_id:integer;
    wait_timer_was:cardinal;
    served:boolean;
    procedure GoToBank(Sender: TObject);
    procedure WaitTime(Sender: TObject);
    procedure FreeTerm(Sender: TObject);
  public
    be_in_queue:boolean;
    mu_time:cardinal;
    wait_time:cardinal;
    operation_sum:integer;
    wait_timer:TTimer;
    day:integer;
    hour:integer;
    constructor Create;
  end;
  TBank=class
  private

```



```

terminals:array of boolean;
term_money_amount:array of integer;
procedure Recharge(Sender: TObject);
procedure AfterRecharge(Sender: TObject);
public
  recharge_timer:TTimer;
  critical_amount:integer;
  recharging:boolean;
  pre_charging:boolean;
  constructor Create;
end;
TQueue=class
  private
    Clients:array of PClient;
    array_length:integer;
  public
    clients_count:integer;
    procedure ReleaseClient(term_id:integer);
    procedure InsertClient(client:PClient);
    procedure StopQueue;
    procedure ReleaseQueue;
    constructor Create;
end;
TCalc=class
  private
    deadline:cardinal;
    step_calc:cardinal;
    avg_serve_time:real;
    queue_time:array of cardinal;
    queue_time_length:cardinal;
    avg_queue_time:real;
    queue_quan:array of cardinal;
    queue_quan_length:integer;
    avg_queue_quan:real;
    procedure Step(Sender: TObject);
  public
    freq_timer:TTimer;
    welcome_timer:TTimer;
    rest_recharge:array of cardinal;
    count_recharge:byte;
    avg_rest_recharge:real;
    n_serv_qtb:integer;
    n_serv_rech:integer;
    n_serv_nem:integer;
    n_serv_rq:integer;
    welcome_count:integer;
    empty_count:integer;
    q,A:real;
    procedure AvgServeTime;
    procedure AvgQueueTime;
    procedure AvgQueueQuan;
    procedure RelAbsTraf;
    procedure ServedClients;

```



```

    procedure WelcomeCheck(Sender: TObject);
    procedure AvgRestRecharge(Sender: TObject);
    constructor Create;
end;
TfrmMain = class(TForm)
    Memol: TMemo;
    Button1: TButton;
    txtTerminals: TEdit;
    txtCountdown: TEdit;
    Label5: TLabel;
    Label6: TLabel;
    txtCl_count: TEdit;
    Label7: TLabel;
    GroupBox1: TGroupBox;
    Label4: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    Label10: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    Label15: TLabel;
    Label18: TLabel;
    Label23: TLabel;
    Label16: TLabel;
    Label17: TLabel;
    StringGrid1: TStringGrid;
    StringGrid2: TStringGrid;
    txtTermMoney: TEdit;
    txtCritical: TEdit;
    txtQueueLimit: TEdit;
    txtEta: TEdit;
    txtTeta: TEdit;
    txtRo: TEdit;
    txtAlpha: TEdit;
    Label2: TLabel;
    txtTime: TEdit;
    Button2: TButton;
    Button3: TButton;
    MainMenu1: TMainMenu;
    N1: TMenuItem;
    N2: TMenuItem;
    N3: TMenuItem;
    N4: TMenuItem;
    N6: TMenuItem;
    txtRealiseQuant: TEdit;
    Label1: TLabel;
    txtNumRealis: TEdit;
    Label3: TLabel;
    procedure Button1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure N4Click(Sender: TObject);

```



```

    procedure N2Click(Sender: TObject);
    procedure N3Click(Sender: TObject);
    procedure N6Click(Sender: TObject);
    procedure FreeAll(Sender: TObject);
private
    { Private declarations }
public
    free_all_timer:TTimer;
    free_all_flag:boolean;
    global_iter_count:integer;
    welcome:array of real;
    empty:array of real;
    recharge:array of real;
    numR:integer;
    { Public declarations }
end;

var
    frmMain: TfrmMain;
    World:TWorld;
    Bank:TBank;
    Queue:TQueue;
    Calc:TCalc;
implementation
{$R *.dfm}
//=====
//===== World implement begin =====
constructor TWorld.Create;
begin
    inherited Create;
end;
function TWorld.LogNormal:real;
var sum1_12, gamma: real;
    l: integer;
begin
    sum1_12:=0;
    for l:=1 to 12 do
        begin
            gamma:=random;
            sum1_12:=sum1_12+gamma;
        end;
    LogNormal:=exp(ln(ro)+alpha*(sum1_12-6));
end;

function TWorld.Gamma:real;
var sum1_eta: real;
    i: Integer;
begin
    sum1_eta:=0;
    for i:=1 to Round(eta) do
        sum1_eta:=sum1_eta+ln(1-random);
    Result:=-teta*sum1_eta;
end;

```




```

procedure TWorld.Start;
begin
  Bank:=TBank.Create;
  Queue:=TQueue.Create;
  Calc:=TCalc.Create;
  ro:=strtofloat(frmMain.txtEta.Text);
  alpha:=strtofloat(frmMain.txtTeta.Text);
  eta:=strtofloat(frmMain.txtRo.Text);
  teta:=strtofloat(frmMain.txtAlpha.Text);
  accumulate_time:=0;
  clients_count:=0;
  served_clients:=0;
  randomize;
  CreateClients;
  Calc.welcome_timer.Enabled:=true;
  StartTimers;
  Calc.freq_timer.Enabled:=true;
  frmMain.free_all_timer.Enabled:=true;
end;
procedure TWorld.CreateClients;
var i,j,k,l:integer;
    intens_per_hour, random_gener,exp_gener,exp_gener_mu:real;
    exp_gener_int,exp_gener_int_mu:int64;
begin
  i:=0;
  l:=0;
  for k:=0 to 4 do
  begin
    for j:=0 to 23 do
    begin
      intens_per_hour:=strtofloat(frmMain.StringGrid1.Cells[0,k])*
      strtofloat(frmMain.StringGrid2.Cells[0,j]);
      l:=l+100;
      while accumulate_time<l do
      begin
        setlength(clients,i+1);           // \
        new(clients[i]);                 // \
        clients[i]^:=TClient.Create;     //  -> Create new client
        clients[i].me:=clients[i];       // /
        clients[i].id:=i+1;              // /
        clients[i].served:=false;        //
        random_gener:=random;
        if random_gener=0 then random_gener:=0.01;
        exp_gener:=(-1/intens_per_hour)*ln(random_gener);
        exp_gener:=exp_gener*100;
        exp_gener_mu:=Gamma;
        exp_gener_mu:=exp_gener_mu*(100/60);
        exp_gener_int:=round(exp_gener);
        exp_gener_int_mu:=round(exp_gener_mu);
        if exp_gener_int_mu=0 then exp_gener_int_mu:=1;
        clients[i].wait_timer.Interval:=accumulate_time+exp_gener_int;
        clients[i].mu_time:=exp_gener_int_mu;
      end;
    end;
  end;
end;

```



```

        clients[i].operation_sum:=round(LogNormal);
        if clients[i].operation_sum > 2000 then
clients[i].operation_sum:= 2000;
        clients[i].hour:=j+1;
        clients[i].day:=k+1;
        i:=i+1;
        clients_count:=clients_count+1;
        accumulate_time:=accumulate_time+exp_gener_int;
    end;
end;
end;
frmMain.txtCl_count.Text:=inttostr(clients_count-1);
setlength(Calc.queue_time,clients_count-1);
setlength(Calc.queue_quan,clients_count-1);
end;
procedure TWorld.StartTimers;
var i:integer;
begin
    for i:=0 to clients_count-2 do
        clients[i].wait_timer.Enabled:=true;
    end;
//===== World implement end =====
//===== Client implement begin =====
constructor TClient.Create;
begin
    inherited Create;
    wait_time:=0;
    wait_timer_was:=0;
    be_in_queue:=false;
    wait_timer:=TTimer.Create(nil);
    wait_timer.Enabled:=false;
    wait_timer.OnTimer:=GoToBank;
end;
procedure TClient.GoToBank(Sender: TObject);
var i:integer;
begin
    if Bank.recharging=true then
    begin
        wait_timer.Enabled:=false;
        Calc.n_serv_rech:=Calc.n_serv_rech+1;
        frmMain.Mem1.Lines.Add(inttostr(id)+' ==== RECHARGING, GO
OUT ====');
        exit;
    end;
    if Queue.clients_count >= strtoint(frmMain.txtQueueLimit.Text)
    then
    begin
        wait_timer.Enabled:=false;
        Calc.n_serv_qtb:=Calc.n_serv_qtb+1;
        frmMain.Mem1.Lines.Add(inttostr(id)+' ==== QUEUE IS TOO
BIG, GO OUT ====');
        exit;
    end;
end;

```

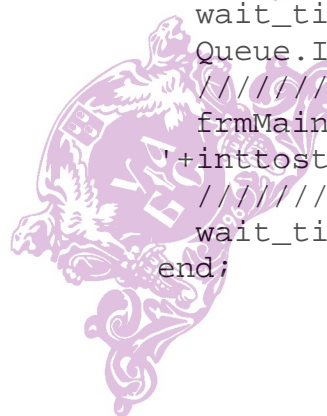


```

if
(Bank.term_money_amount[0]<>0)and(Bank.term_money_amount[0]<oper
ation_sum) then
  operation_sum:=Bank.term_money_amount[0];
  if Bank.term_money_amount[0] = 0 then
  begin
    wait_timer.Enabled:=false;
    Calc.n_serv_nem:=Calc.n_serv_nem+1;
    frmMain.Memo1.Lines.Add(inttostr(id)+' ==== NOT ENOUGH
MONEY, GO OUT ====');
    exit;
  end;
  if (Queue.clients_count=0)and(be_in_queue=false) then
  for i:=0 to strtoint(frmMain.txtTerminals.Text)-1 do
  begin
    if Bank.terminals[i]=false then
    begin
      wait_timer.Enabled:=false;
      wait_timer_was:=wait_timer.Interval;
      term_id:=i;
      wait_timer.Interval:=mu_time;
      wait_timer.OnTimer:=FreeTerm;
      Bank.terminals[i]:=true;
      if (Bank.critical_amount >= Bank.term_money_amount[i]-
operation_sum)and(Bank.pre_charging=false) then
      begin
        Bank.pre_charging:=true;
        Bank.recharge_timer.Enabled:=true;
      end;
      Bank.term_money_amount[i]:=Bank.term_money_amount[i]-
operation_sum;
      ////////////////////////////////////////////////////////////////////
      frmMain.Memo1.Lines.Add('# '+inttostr(id)+' '+'day
'+inttostr(day)+' '+'hour '+inttostr(hour)+' DIRECTLY to
terminal '+'serv time '+inttostr(mu_time)+' '+'amount
'+inttostr(operation_sum));
      ////////////////////////////////////////////////////////////////////
      wait_timer.Enabled:=true;
      exit;
    end;
  end;
  wait_timer.Enabled:=false;
  wait_timer_was:=wait_timer.Interval;
  wait_timer.Interval:=1;
  wait_timer.OnTimer:=WaitTime;
  Queue.InsertClient(me);
  ////////////////////////////////////////////////////////////////////
  frmMain.Memo1.Lines.Add('# '+inttostr(id)+' '+'day
'+inttostr(day)+' '+'hour '+inttostr(hour)+' TO QUEUE');
  ////////////////////////////////////////////////////////////////////
  wait_timer.Enabled:=true;
end;

```

State Higher Educational Institution
"UKRAINIAN ACADEMY OF BANKING
OF THE NATIONAL BANK OF UKRAINE"



```

procedure TClient.WaitTime(Sender: TObject);
begin
    wait_time:=wait_time+1;
end;
procedure TClient.FreeTerm(Sender: TObject);
begin
    wait_timer.Enabled:=false;
    Bank.terminals[term_id]:=false;
    served:=true;
    //////////////////////////////////////
    frmMain.Memol.Lines.Add('# '+inttostr(id)+' '+'hour
'+inttostr(wait_timer_was+wait_time+mu_time)+' OUT from
terminal ');
    //////////////////////////////////////
    if Queue.clients_count<>0 then
        Queue.ReleaseClient(term_id);
    inc(World.served_clients);
end;
//===== Client implement end =====
//===== Bank implement begin =====
constructor TBank.Create;
var i:integer;
begin
    inherited Create;
    setlength(terminals,strtoint(frmMain.txtTerminals.Text));
    setlength(term_money_amount,strtoint(frmMain.txtTerminals.Text))
;
    for i:=0 to length(terminals)-1 do
    begin
        terminals[i]:=false;
        term_money_amount[i]:=strtoint(frmMain.txtTermMoney.Text);
    end;
    recharge_timer:=TTimer.Create(nil);
    recharge_timer.Enabled:=false;
    recharge_timer.OnTimer:=Recharge;
    recharge_timer.Interval:=round(10*(100/60));
    critical_amount:=strtoint(frmMain.txtCritical.Text);
    recharging:=false;
    pre_charging:=false;
end;
procedure TBank.Recharge;
begin
    recharging:=true;
    recharge_timer.Enabled:=false;
    Queue.ReleaseQueue;
    recharge_timer.Interval:=round(20*(100/60));
    recharge_timer.OnTimer:=AfterRecharge;
    recharge_timer.Enabled:=true;
    setlength(Calc.rest_recharge,length(Calc.rest_recharge)+1);
    Calc.rest_recharge[length(Calc.rest_recharge)-
1]:=term_money_amount[0];
    inc(Calc.count_recharge);
end;

```




```

procedure TBank.AfterRecharge;
begin
    recharge_timer.Enabled:=false;
    recharge_timer.OnTimer:=Recharge;
    recharge_timer.Interval:=round(10*(100/60));
    term_money_amount[0]:=strtoint(frmMain.txtTermMoney.Text);
    recharging:=false;
    pre_charging:=false;
end;
//===== Bank implement end =====
//===== Queue implement begin =====
constructor TQueue.Create;
begin
    inherited Create;
    clients_count:=0;
    array_length:=1;
end;
procedure TQueue.InsertClient(client:PClient);
begin
    clients_count:=clients_count+1;
    array_length:=array_length+1;
    setlength(clients,array_length);
    clients[array_length-1]:=client;
    client.be_in_queue:=true;
end;
procedure TQueue.ReleaseQueue;
var i:integer;
begin
    frmMain.Memo1.Lines.Add('===== RELEASE QUEUE =====');
    Calc.n_serv_rq:=Queue.clients_count;
    if Queue.clients_count>0 then
        for i:=array_length downto 1 do
            if clients[i-1]<>nil then
                begin
                    clients[i-1].wait_timer.Enabled:=false;
                    clients[i-1]:=nil;
                end;
            clients_count:=0;
        end;
    procedure TQueue.ReleaseClient(term_id:integer);
    var i:integer;
    begin
        for i:=0 to length(clients)-1 do
            if clients[i]<>nil then
                begin
                    clients[i].wait_timer.Enabled:=false;
                    Calc.queue_time[Calc.queue_time_length]:=clients[i].wait_time;
                    inc(Calc.queue_time_length);
                    if
                        (Bank.term_money_amount[0]<>0)and(Bank.term_money_amount[0]<clients[i].operation_sum) then

```



```

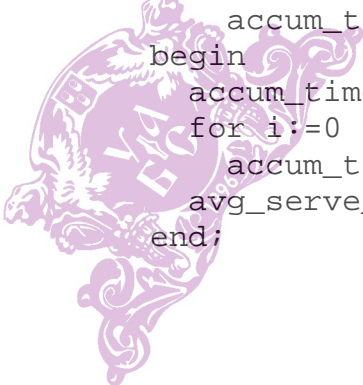
    clients[i].operation_sum:=Bank.term_money_amount[0];
if Bank.term_money_amount[0] = 0 then
begin
    clients[i].wait_timer.Enabled:=false;
    Calc.n_serv_nem:=Calc.n_serv_nem+1;
    frmResult.txtNServe_nem.Text:=inttostr(Calc.n_serv_nem);
    frmMain.Memo1.Lines.Add(inttostr(clients[i].id)+'====
NOT ENOUGH MONEY, GO OUT ====');
    exit;
end;
clients[i].wait_timer.Interval:=clients[i].mu_time;
clients[i].wait_timer.OnTimer:=clients[i].FreeTerm;
clients[i].term_id:=term_id;
Bank.terminals[term_id]:=true;
if (Bank.critical_amount >= Bank.term_money_amount[0]-
clients[i].operation_sum)and(Bank.pre_charging=false) then
begin
    Bank.pre_charging:=true;
    Bank.recharge_timer.Enabled:=true;
end;
Bank.term_money_amount[term_id]:=Bank.term_money_amount[term_id]
-clients[i].operation_sum;
////////////////////////////////////
    frmMain.Memo1.Lines.Add('# '+inttostr(clients[i].id)+'
'+inttostr(clients[i].wait_timer_was+clients[i].wait_time)+'da
y '+inttostr(clients[i].day)+' '+'hour
'+inttostr(clients[i].hour)+' FROM QUEUE to terminal '+' '+'serv
time '+inttostr(clients[i].mu_time)+' '+'amount
'+inttostr(clients[i].operation_sum));
////////////////////////////////////
    clients[i].wait_timer.Enabled:=true;
    clients[i]:=nil;
    clients_count:=clients_count-1;
    exit;
end;
end;
procedure TQueue.StopQueue;
var i:integer;
begin
    for i:=0 to length(World.clients)-1 do
        world.clients[i].wait_timer.Enabled:=false;
end;
//===== Queue implement end =====
//===== Calc implement begin =====
constructor TCalc.Create;
begin
    inherited Create;
    freq_timer:=TTimer.Create(nil);
    freq_timer.Enabled:=false;
    freq_timer.Interval:=100;
    freq_timer.OnTimer:=Step;
    deadline:=strtoint(frmMain.txtTime.Text)*100;
    step_calc:=0;

```

```

queue_time_length:=0;
queue_quan_length:=0;
count_recharge:=0;
n_serv_qtb:=0;
n_serv_rech:=0;
n_serv_nem:=0;
n_serv_rq:=0;
welcome_timer:=TTimer.Create(nil);
welcome_timer.Enabled:=false;
welcome_timer.Interval:=1;
welcome_timer.OnTimer:=WelcomeCheck;
welcome_count:=0;
empty_count:=0;
end;
procedure TCalc.Step(Sender: TObject);
begin
step_calc:=step_calc+100;
frmMain.txtCountdown.Text:=floattostr(step_calc/100);
queue_quan[queue_quan_length]:=Queue.clients_count;
inc(queue_quan_length);
if step_calc=deadline then
begin
freq_timer.Enabled:=false;
Queue.StopQueue;
welcome_timer.Enabled:=false;
ServedClients;
AvgServeTime;
AvgQueueTime;
AvgQueueQuan;
RelAbsTraf;
AvgRestRecharge(Sender);
////////////////////////////////////
frmMain.free_all_flag:=true;
end;
end;
procedure TCalc.ServedClients;
var i,served:integer;
begin
served:=0;
for i:=0 to World.clients_count-2 do
if World.clients[i].served=true then
inc(served);
end;
procedure TCalc.AvgServeTime;
var i:integer;
accum_time:cardinal;
begin
accum_time:=0;
for i:=0 to World.clients_count-2 do
accum_time:=accum_time+World.clients[i].mu_time;
avg_serve_time:=accum_time/(World.clients_count-1);
end;

```



```

procedure TCalc.AvgQueueTime;
var i:integer;
    accum_time:cardinal;
begin
    accum_time:=0;
    for i:=0 to queue_time_length-1 do
        accum_time:=accum_time+queue_time[i];
    if queue_time_length<>0 then
        avg_queue_time:=accum_time/queue_time_length
    else
        avg_queue_time:=0;
    end;
procedure TCalc.AvgQueueQuan;
var i:integer;
    accum_quan:cardinal;
begin
    accum_quan:=0;
    for i:=0 to World.clients_count-2 do
        if World.clients[i].be_in_queue=true then inc(accum_quan);
    avg_queue_quan:=accum_quan/(queue_quan_length);
    avg_queue_quan:=roundto(avg_queue_quan,-2);
end;
procedure TCalc.RelAbsTraf;
var p_otk:real;
begin
    p_otk:=(World.clients_count-
World.served_clients)/strtoint(frmMain.txtCl_count.Text);
    q:=1-p_otk;
    A:=((World.clients_count-1)/strtoint(frmMain.txtTime.Text))*q;
    q:=roundto(q,-3);
    A:=roundto(A,-3);
end;
procedure TCalc.WelcomeCheck(Sender: TObject);
begin
    if (Bank.recharging=false)and(Bank.term_money_amount[0]=0)
then
    begin
        inc(empty_count);
        exit;
    end;
    if Bank.terminals[0]=false then inc(welcome_count);
end;
procedure TCalc.AvgRestRecharge;
var u, avg:cardinal;
begin
    avg:=0;
    if Calc.count_recharge<>0 then
    begin
        for u:=0 to Calc.count_recharge-1 do
            avg:=avg+Calc.rest_recharge[u];
        Calc.avg_rest_recharge:=avg/Calc.count_recharge;
    end
    else

```




```

    Calc.avg_rest_recharge:=0;
end;
//===== Calc implement end =====
procedure TfrmMain.Button1Click(Sender: TObject);
var r:integer;
    temp1, temp2, temp3, temp4, temp5, temp6, temp7, temp8,
temp9, temp10, temp11, temp12, temp13, temp14, temp15, temp16,
tempK1, tempK2, tempK3, tempK4:real;
begin
setlength(frmMain.welcome,strtoint(frmMain.txtRealiseQuant.Text)
);
setlength(frmMain.empty,strtoint(frmMain.txtRealiseQuant.Text));
setlength(frmMain.recharge,strtoint(frmMain.txtRealiseQuant.Text
));
    if
frmMain.global_iter_count<strtoint(frmMain.txtRealiseQuant.Text)
then
    begin
        frmMain.free_all_flag:=false;
        frmMain.Memo1.Lines.Clear;
        World:=TWorld.Create;
        World.Start;
        inc(frmMain.numR);
        frmMain.txtNumRealis.Text:=inttostr(numR);
    end
    else if
frmMain.global_iter_count=strtoint(frmMain.txtRealiseQuant.Text)
then
    begin
        showmessage('Моделювання закінчено!');
        frmMain.numR:=0;
        frmMain.global_iter_count:=0;
        temp1:=0;
        temp2:=0;
        temp3:=0;
        temp4:=0;
        temp5:=0;
        temp6:=0;
        temp7:=0;
        temp8:=0;
        temp9:=0;
        temp10:=0;
        temp11:=0;
        temp12:=0;
        temp13:=0;
        temp14:=0;
        temp15:=0;
        temp16:=0;
        tempK1:=0;
        tempK2:=0;
        tempK3:=0;
        for r:=0 to strtoint(frmMain.txtRealiseQuant.Text)-1 do
        begin

```



```

temp1:=temp1+strtof(float(frmResult.StringGrid1.Cells[r,1]));
temp2:=temp2+strtof(float(frmResult.StringGrid1.Cells[r,2]));
temp3:=temp3+strtof(float(frmResult.StringGrid1.Cells[r,3]));
temp4:=temp4+strtof(float(frmResult.StringGrid1.Cells[r,4]));
temp5:=temp5+strtof(float(frmResult.StringGrid1.Cells[r,5]));
temp6:=temp6+strtof(float(frmResult.StringGrid1.Cells[r,6]));
temp7:=temp7+strtof(float(frmResult.StringGrid1.Cells[r,7]));
temp8:=temp8+strtof(float(frmResult.StringGrid1.Cells[r,8]));
temp9:=temp9+strtof(float(frmResult.StringGrid1.Cells[r,9]));
temp10:=temp10+strtof(float(frmResult.StringGrid1.Cells[r,10]));
temp11:=temp11+strtof(float(frmResult.StringGrid1.Cells[r,11]));
temp12:=temp12+strtof(float(frmResult.StringGrid1.Cells[r,12]));
temp13:=temp13+strtof(float(frmResult.StringGrid1.Cells[r,13]));
temp14:=temp14+strtof(float(frmResult.StringGrid1.Cells[r,14]));
temp15:=temp15+strtof(float(frmResult.StringGrid1.Cells[r,15]));
temp16:=temp16+strtof(float(frmResult.StringGrid1.Cells[r,16]));
tempK1:=tempK1+welcome[r];
tempK2:=tempK2+empty[r];
tempK3:=tempK3+recharge[r];
end;
frmResult.txtClCount.Text:=floattostr(roundto(temp1/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtServe.Text:=floattostr(roundto(temp2/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtNoServe.Text:=floattostr(roundto(temp3/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtNServe_nem.Text:=floattostr(roundto(temp4/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtNServ_rq.Text:=floattostr(roundto(temp5/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtNServ_rech.Text:=floattostr(roundto(temp6/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtNServ_qtb.Text:=floattostr(roundto(temp7/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtAvgQueueQuan.Text:=floattostr(roundto(temp8/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtAvgQueueTime.Text:=floattostr(roundto(temp9/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtAvgServeTime.Text:=floattostr(roundto(temp10/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtRechargePeriod.Text:=floattostr(roundto(temp11/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtRestRecharge.Text:=floattostr(roundto(temp12/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtAbsTraf.Text:=floattostr(roundto(temp13/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtRelTraf.Text:=floattostr(roundto(temp14/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtLambdaEmp.Text:=floattostr(roundto(temp15/strtoint(frmMain.txtRealiseQuant.Text),-3));
frmResult.txtMuEmp.Text:=floattostr(roundto(temp16/strtoint(frmMain.txtRealiseQuant.Text),-3));
//////////-----RESULTS END-----//////////

```

```

//////////-----KOEFFICIENTS-----//////////
frmResult.txtKUse.Text:=floattostr(roundto((strtofloat(frmResult
.txtServe.Text)*strtofloat(frmResult.txtAvgServeTime.Text)/(60*1
20)),-3));
frmResult.txtKWait.Text:=floattostr(roundto((tempK1/120),-
3));
frmResult.txtKEmpty.Text:=floattostr(roundto((tempK2/120),-
3));
frmResult.txtKRecharge.Text:=floattostr(roundto((tempK3/120),-
3));
//////////-----KOEFFICIENTS END-----//////////
end;
end;
procedure TfrmMain.FormCreate(Sender: TObject);
var i: integer;
begin
frmMain.numR:=0;
free_all_flag:=false;
global_iter_count:=0;
//////////
StringGrid1.Cells[0,0]:='85,5';
StringGrid1.Cells[0,1]:='74,25';
StringGrid1.Cells[0,2]:='114,75';
StringGrid1.Cells[0,3]:='131,4';
StringGrid1.Cells[0,4]:='255,75';
//////////
StringGrid2.Cells[0,0]:='0,005';
StringGrid2.Cells[0,1]:='0,005';
StringGrid2.Cells[0,2]:='0,005';
StringGrid2.Cells[0,3]:='0,005';
StringGrid2.Cells[0,4]:='0,005';
StringGrid2.Cells[0,5]:='0,006';
StringGrid2.Cells[0,6]:='0,016';
StringGrid2.Cells[0,7]:='0,027';
StringGrid2.Cells[0,8]:='0,093';
StringGrid2.Cells[0,9]:='0,053';
StringGrid2.Cells[0,10]:='0,055';
StringGrid2.Cells[0,11]:='0,055';
StringGrid2.Cells[0,12]:='0,019';
StringGrid2.Cells[0,13]:='0,110';
StringGrid2.Cells[0,14]:='0,058';
StringGrid2.Cells[0,15]:='0,079';
StringGrid2.Cells[0,16]:='0,097';
StringGrid2.Cells[0,17]:='0,161';
StringGrid2.Cells[0,18]:='0,077';
StringGrid2.Cells[0,19]:='0,027';
StringGrid2.Cells[0,20]:='0,014';
StringGrid2.Cells[0,21]:='0,011';
StringGrid2.Cells[0,22]:='0,007';
StringGrid2.Cells[0,23]:='0,010';
free_all_timer:=TTimer.Create(nil);
free_all_timer.Interval:=100;
free_all_timer.OnTimer:=FreeAll;

```




```

end;
procedure TfrmMain.FreeAll(Sender: TObject);
var h:integer;
begin
  if frmMain.free_all_flag=true then
  begin
    frmMain.free_all_timer.Enabled:=false;
    for h:=0 to World.clients_count-1 do
    begin
      world.clients[h].wait_timer.Free;
      world.clients[h]^Free;
      dispose(world.clients[h]);
    end;
    Queue.Free;
    Bank.recharge_timer.Free;
    Bank.Free;
    Calc.freq_timer.Free;
    Calc.welcome_timer.Free;
    //////////////////////////////////---RESULTS---////////////////////////////////
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,0]:=inttostr(
    frmMain.numR);
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,1]:=floattostr(
    strtoint(frmMain.txtCl_count.Text)/5);
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,2]:=floattostr(
    World.served_clients/5);
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,3]:=floattostr(
    (strtoint(frmMain.txtCl_count.Text)-
    World.served_clients)/5);
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,4]:=floattostr(
    ostr(Calc.n_serv_nem/5));
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,5]:=floattostr(
    ostr(Calc.n_serv_rq/5));
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,6]:=floattostr(
    ostr(Calc.n_serv_rech/5));
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,7]:=floattostr(
    ostr(Calc.n_serv_qtb/5));
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,8]:=floattostr(
    ostr(Calc.avg_queue_quan));
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,9]:=floattostr(
    ostr(roundto(Calc.avg_queue_time/(100/60),-3)));
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,10]:=floattostr(
    ostr(roundto(Calc.avg_serve_time/(100/60),-3)));
    if Calc.count_recharge<>0 then
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,11]:=floattostr(
    ostr(120/Calc.count_recharge))
    else
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,11]:='120'
    ;
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,12]:=floattostr(
    ostr(Calc.avg_rest_recharge));
    frmResult.StringGrid1.Cells[frmMain.global_iter_count,13]:=floattostr(
    ostr(Calc.A));
  end;
end;

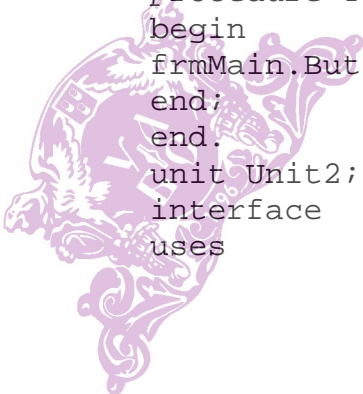
```




```

frmResult.StringGrid1.Cells[frmMain.global_iter_count,14]:=float
tostr( Calc.q );
frmResult.StringGrid1.Cells[frmMain.global_iter_count,15]:=float
tostr(roundto((World.clients_count-
1)/strtoint(frmMain.txtTime.Text),-2));
frmResult.StringGrid1.Cells[frmMain.global_iter_count,16]:=float
tostr(roundto(World.served_clients/strtoint(frmMain.txtTime.Text
),-2));
/////-----////////////////////////////////////
frmMain.welcome[global_iter_count]:=roundto((Calc.welcome_count/
(100/60))/60,-3);
frmMain.empty[global_iter_count]:=roundto((Calc.empty_count/(100
/60))/60,-3);
frmMain.recharge[global_iter_count]:=roundto((Calc.count_recharg
e*20/60),-3);
////////////////////////////////-----RESULTS-----////////////////////////////////
Calc.Free;
World.Free;
inc(frmMain.global_iter_count);
frmMain.Button1.Click;
end;
end;
procedure TfrmMain.Button2Click(Sender: TObject);
begin
frmResult.Show;
end;
procedure TfrmMain.Button3Click(Sender: TObject);
begin
frmMain.Close;
frmResult.Close;
end;
procedure TfrmMain.N4Click(Sender: TObject);
begin
Application.HelpFile:='Help.hlp';
Application.HelpCommand(Help_Finder, 0);
end;
procedure TfrmMain.N2Click(Sender: TObject);
begin
frmMain.Button1.Click;
end;
procedure TfrmMain.N3Click(Sender: TObject);
begin
frmMain.Button2.Click;
end;
procedure TfrmMain.N6Click(Sender: TObject);
begin
frmMain.Button3.Click;
end;
end.
unit Unit2;
interface
uses

```



```
Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms,
Dialogs, StdCtrls, Grids;
type
```

```
TfrmResult = class(TForm)
  txtAbsTraf: TEdit;
  txtRelTraf: TEdit;
  txtLambdaEmp: TEdit;
  txtMuEmp: TEdit;
  txtAvgServeTime: TEdit;
  txtAvgQueueQuan: TEdit;
  txtAvgQueueTime: TEdit;
  txtNoServe: TEdit;
  Label18: TLabel;
  Label17: TLabel;
  Label15: TLabel;
  Label16: TLabel;
  Label10: TLabel;
  Label14: TLabel;
  Label13: TLabel;
  Label8: TLabel;
  Button1: TButton;
  txtNServe_nem: TEdit;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  txtNServ_rech: TEdit;
  txtNServ_qtb: TEdit;
  Label5: TLabel;
  Label6: TLabel;
  txtNServ_rq: TEdit;
  StringGrid1: TStringGrid;
  Label7: TLabel;
  Label9: TLabel;
  Label11: TLabel;
  Label12: TLabel;
  txtClCount: TEdit;
  txtServe: TEdit;
  txtRechargePeriod: TEdit;
  txtRestRecharge: TEdit;
  Label19: TLabel;
  Label20: TLabel;
  Label21: TLabel;
  Label22: TLabel;
  txtKUse: TEdit;
  txtKWait: TEdit;
  txtKEmpty: TEdit;
  txtKRecharge: TEdit;
  Button2: TButton;
  Label23: TLabel;
  procedure Button1Click(Sender: TObject);
  procedure Button2Click(Sender: TObject);
```



```
private
  { Private declarations }
public
  { Public declarations }
end;
var
  frmResult: TfrmResult;
implementation
uses Unit1;
{$R *.dfm}
procedure TfrmResult.Button1Click(Sender: TObject);
begin
  frmResult.Hide;
end;
procedure TfrmResult.Button2Click(Sender: TObject);
var asd: textfile;
    i, j: integer;
    our_row: string;
begin
  assignfile(asd, 'asd.txt');
  rewrite(asd);
  for i:=0 to 16 do
  begin
    for j:=0 to strtoint(frmMain.txtRealiseQuant.text)-1 do
      our_row:=our_row+' '+frmResult.StringGrid1.Cells[j,i];
    writeln(asd,our_row);
    our_row:='';
  end;
  closefile(asd);
end;
end.
```

Державний вищий навчальний заклад
"УКРАЇНСЬКА АКАДЕМІЯ БАНКІВСЬКОЇ СПРАВИ
НАЦІОНАЛЬНОГО БАНКУ УКРАЇНИ"

State Higher Educational Institution
"UKRAINIAN ACADEMY OF BANKING
OF THE NATIONAL BANK OF UKRAINE"

