

# МОДЕЛЮВАННЯ І АВТОМАТИЗАЦІЯ ОЦІНКИ ОПЕРАЦІЙНИХ РИЗИКІВ В КОМЕРЦІЙНИХ БАНКАХ

Конкурсна робота Всеукраїнського  
конкурсу студентських наукових робіт з  
природничих, технічних та гуманітарних наук у 2011/2012 навчальному році

Виконавець – студент 5-го курсу,  
спеціальності «Економічна кібернетика»  
Державного вищого навчального закладу «Українська  
академія банківської справи Національного банку  
України»  
Радовіченко Олексій Олегович

Керівник – к.е.н., доцент, доцент кафедри  
економічної кібернетики Державного вищого  
навчального закладу «Українська академія  
банківської справи Національного банку України»  
Яровенко Ганна Миколаївна

## ЗМІСТ

ВСТУП.....	3
1 МОДЕЛЬ ОЦІНКИ ОПЕРАЦІЙНИХ РИЗИКІВ В КОМЕРЦІЙНИХ БАНКАХ.....	5
2 РОЗРОБКА ПРОГРАМНОГО ДОДАТКУ ДЛЯ РЕАЛІЗАЦІЇ МОДЕЛІ .....	15
3 АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ ТА ПЕРЕВІРКА МОДЕЛІ НА ЧУТЛИВІСТЬ.....	21
4 ЕФЕКТ ВІД ВПРОВАДЖЕННЯ ЗАПРОПОНОВАНОЇ МОДЕЛІ ТА ПРОГРАМНОГО ДОДАТКУ .....	25
ВИСНОВКИ .....	29
ПЕРЕЛІК ПОСИЛАНЬ .....	30
ДОДАТКИ .....	31

## ВСТУП

Протягом останнього часу в світовій банківській практиці приділяють значну увагу операційним ризикам (далі ОР). Першопричиною такої уваги є прямі та побічні збитки, породжені операційною діяльністю комерційних банків, яка нині набула якісно нових ознак, пов'язаних із масштабами операцій банків, їх різноманітністю, запровадженням нових банківських технологій тощо.

В процесі удосконалення та спрощення системи контролю банківського сектору відбувається перехід до міжнародних стандартів, описаних в Базельських угодах. На сучасному етапі банки працюють за «Базелем II», але вже досить довго говорять про необхідність переходу до «Базеля III». Було розроблено велику документальну базу стосовно основних положень «Базеля III», в яких є зміни стосовно операційного ризику. Переоцінюється саме поняття операційного ризику та його складових. Це пов'язано із зростаючою долею ризиків в період кризи, та післякризове відновлення.

Щоб домогтись відповідності банківської сфери міжнародним стандартам, слід починати вже сьогодні готуватись до змін в «Базелі III». Тому основною метою роботи є розробка моделі для оцінки операційного ризику та його компонент в розрізі саме цього стандарту, а також розробка програмного додатку для реалізації моделі.

За основу моделі визначення операційного ризику комерційного банку обрано нечітко-множинний та імовірнісний (Баєсівський) підходи. Математична модель повинна відповідати таким вимогам:

- відображати суттєві риси процесу управління операційним ризиком банку і при цьому не сильно його спрощувати;
- бути адекватною стосовно досліджуваного процесу;
- використовуватись для прийняття управлінських рішень;
- надавати як якісну, так і кількісну оцінку управління операційним ризиком по конкретному банку;

– дозволяти виділити найбільш небезпечні фактори ризику і їх імовірність настання.

Оскільки спеціальних програмних додатків для реалізації подібної моделі не існує, то однією із задач даної роботи є також розробка такого додатку, який дозволить отримувати оцінки операційних ризиків та рівень втрат від них.

# 1 МОДЕЛЬ ОЦІНКИ ОПЕРАЦІЙНИХ РИЗИКІВ В КОМЕРЦІЙНИХ БАНКАХ

Основою моделі визначення операційного ризику комерційного банку виступає комбінація нечітко-множинного та імовірнісного (Баєсівського) підходів.

Реалізація першого етапу запропонованої моделі - визначення кількісної оцінки ступеня операційного ризику.

Другим етапом моделі виступає проведення експрес-оцінки ОР за чотирма інцидентами, що пропонує Національний банк України. Але в зв'язку з втрачанням актуальності «Базеля II» і необхідністю більш детального аналізу ОР за «Базелем III», слід розраховувати експрес-оцінки за сімома інцидентами. Тому пропонується визначати суму бінарних характеристик за всіма фінансовими показниками та в рамках семи виділених інцидентів (формула 1.1):

$$EO_m = \sum_{i=1}^n \sum_{j=1}^7 Kb_{ijm}, \quad (1.1)$$

де  $EO_m$  - експрес-оцінка  $m$ -го комерційного банку ступеня його ОР;

$n$  - кількість фінансових показників - індикаторів прямих та опосередкованих наслідків виявленого операційного ризику;

$Kb_{ijm}$  - бінарна характеристика в розрізі  $i$ -го ( $i=1 \div n$ ) фінансового показника,  $j$ -го ( $j=1 \div 7$ ) інциденту  $m$ -го комерційного банку ступеня його ОР.

За результатами розрахованої експрес-оцінки пропонується провести групування банківських установ за самостійно розробленими інтервалами:

- якщо отримана експрес-оцінка належить проміжку від  $\min\{EO_m\}$  до  $\frac{2 * \max\{EO_m\}}{7}$ , рівень його операційного ризику є нормальним;

– якщо належить проміжку від  $\frac{2 * \max\{EO_m\}}{7} \leq EO_m \leq \frac{3 * \max\{EO_m\}}{7}$  - допустимий рівень ризику;

– у випадку відповідності проміжку  $\frac{4 * \max\{EO_m\}}{7} \leq EO_m \leq \frac{5 * \max\{EO_m\}}{7}$  - високий рівень операційного ризику;

– для проміжку від рівня  $\frac{6 * \max\{EO_m\}}{7}$  до  $\max\{EO_m\}$  - ставиться у відповідність критичний рівень ризику.

Логічним продовженням реалізації експрес-підходу до оцінки відповідності фінансових показників визначеним інцидентам ОР виступає реалізація третього етапу – додаткового проведення більш глибокого, детального та структурного аналізу – комплексного підходу оцінки операційного ризику. Сутність даного підходу полягає у розрахунку зважених фінансових показників на рівень відповідності певним інцидентам ОР, при чому ваги визначаються на основі причинних факторів.

Результатом застосування другого і третього етапів запропонованого методичного підходу оцінки операційного ризику в розрізі кожного конкретного банку виступає таблиця відповідності фінансових показників визначеним інцидентам операційного ризику (див. додаток А).

Реалізація п'ятого етапу передбачає визначення оцінки ступеня операційного ризику комерційного банку. Пропонується сформуванати групу показників  $K_{ij}, i = 1 \div n, j = 1 \div m$ , кожен із яких у певній мірі характеризує той чи інший  $j$ -й інцидент (причину) виникнення операційного ризику.

На основі того, що показники, які характеризують рівень операційного ризику, відображають різні аспекти функціонування банку і відповідно є різнорідними, необхідно привести їх у співставний вигляд (визначити нормалізоване значення).

З метою визначення числових значень характеристик ступеня впливу певного інциденту на рівень показника операційного ризику проведено

наступний аналіз, з врахуванням змін, котрі потребує «Базель III» (формула 1.2). Варто зазначити, що показники операційного ризику відображають кожний інцидент ризику у відповідних співвідношеннях. Для проведення даного аналізу представимо групи інцидентів операційного ризику в якості бінарних змінних, тобто змінних, які приймають значення «1» у випадку можливості їх опису відповідним показником, або «0» в іншому випадку.

$$K_i = \beta_0 + \beta_1 F_{1i} + \beta_2 F_{2i} + \beta_3 F_{3i} + \dots + \beta_7 F_{7i} + \varepsilon, \quad (1.2)$$

де  $K_i$  - абсолютне значення  $i$ -го показника кількісної оцінки ступеня ОР;  
 $F_{ij}, j=1 \div 7$  - бінарна характеристика наявності  $j$ -го інциденту операційного ризику зумовленого  $i$ -м показником;  
 $\beta_m, m=0 \div 7$  - сталі величини;  
 $\varepsilon$  - похибка (відхилення фактичного і теоретичного рівнів відповідного  $i$ -го показника кількісної оцінки ступеня операційного ризику).

Визначити числові значення характеристик ступеня впливу певного інциденту на рівень показника операційного ризику до  $j$ -х інцидентів на основі рівняння 1.2 виявляється неможливим. Тому, для того щоб визначити на скільки відсотків кожен з інцидентів пояснює виникнення операційного ризику за відповідним показником (формула 1.3):

$$K_i = \alpha_1 F_{1i} + \alpha_2 F_{2i} + \alpha_3 F_{3i} + \dots + \alpha_7 F_{7i} + \varepsilon, \quad (1.3)$$

де  $\alpha_m, m=1 \div 7$  - сталі величини, які відображають значення характеристик ступеня впливу певного інциденту на рівень показника ОР до  $j$ -х інцидентів.

Коефіцієнти  $\alpha_m, m=1 \div 7$  рівняння 1.3 знаходяться за формулою 1.4:

$$\alpha_m = \beta_m \frac{\sigma_{F_j}}{\sigma_{K_i}}, \quad (1.4)$$

де  $\sigma_{F_j}$ ,  $\sigma_{K_i}$  - середні квадратичні відхилення факторних і результативної ознак відповідно.

В результаті того, що метою дослідження є встановлення абсолютного значення ступеня впливу інцидентів на показники ОР, то отримані значення, в разі невідповідності знаків, беруться по модулю. На основі скорегованих числових характеристик ( $\alpha_m^*$ ) розраховується відносний показник структури, який відображає питому вагу впливу інцидентів на рівень ОР (формула 1.5):

$$\alpha_m^* = \frac{\alpha_m}{\sum_{m=1}^7 \alpha_m} \quad (1.5)$$

Далі відбувається оцінка допустимих (граничних) значень для виявлених нормалізованих показників, зважених на відповідне значення характеристик ступеня впливу певного інциденту на рівень кожного з показників кількісної оцінки ступеня операційного ризику (формування «коридору» допустимих значень нормалізованих показників). Для цього визначимо оптимістичний і песимістичний варіанти нормованих показників кількісної оцінки ступеня ОР банку, враховуючи, що кожен показник може приймати будь-яке значення з діапазону  $0 \div NK_i$ , де  $i=1 \div n$ . Це пояснюється тим, що для оптимістичного варіанту характеристика ступеня впливу певного інциденту приймає значення «0» – ризик відсутній, а для песимістичного варіанту значення «1», тобто ОР не лише присутній, але й приймає максимально можливе значення.

На основі отриманого діапазону допустимих значень нормалізованих показників визначається рівні кількісної оцінки ступеня операційного ризику банку за кожним показником:

- якщо  $0 \leq \alpha_m^* NK_i < 0,3NK_i$ , нормальний рівень;
- якщо  $0,3NK_i \leq \alpha_m^* NK_i < 0,5NK_i$ , підвищений рівень;



– якщо  $0,5NK_i \leq \alpha_m^* NK_i < 0,7NK_i$ , високий рівень;

– якщо  $0,7NK_i \leq \alpha_m^* NK_i \leq NK_i$ , критичний рівень.

Враховуючи приведену вище класифікацію, можна зробити висновок, що допустимим (граничним) рівнем для виявлених нормалізованих показників, зважених на відповідне значення характеристик ступеня впливу певного інциденту, виступає діапазон  $0 \leq \alpha_m^* NK_i < 0,3NK_i$ .

Для визначення бінарних характеристик за нормалізованими показниками  $NK_i, i = 1 \div n$  скористаємось формулою 1.6:

$$NKbin_i \begin{cases} = 1; \alpha_m^* \overline{NK_m} \geq \alpha_m^* NK_i, \\ = 0; \alpha_m^* NK_i < \alpha_m^* \overline{NK_m}, \end{cases} \quad (1.6)$$

де  $NKbin_i$  - бінарні характеристики по кожному показнику кількісної оцінки ступеня операційного ризику банку відповідно до інцидентів даного ризику;

$NK_i, i = 1 \div n$  - нормалізоване значення  $i$ -го показника кількісної оцінки ступеня операційного ризику;

$\overline{NK_m}$  - середнє значення за всіма нормалізованими показниками  $m$ -го інциденту ризику.

Далі розраховується сума бінарних показників для кожного  $j$ -го фактору ризику, які отримали значення «1», тобто експрес-оцінка операційного ризику за  $j$ -м фактором ризику (формула 1.7):

$$EO_j = \sum_{i=1}^n NKbin_{ij}, \quad (1.7)$$

де  $EO_j$  - експрес-оцінка операційного ризику за  $j$ -м фактором ризику.

На основі розрахованої суми бінарних показників для кожного  $j$ -го інциденту ризику визначається загальна сума бінарних показників, яка і виступає експрес-оцінкою операційного ризику банку .

На основі отриманих сум бінарних показників ( $EO$ ), яка виступає кількісною експрес-оцінкою ступеня ОР, визначається якісна оцінка його рівня.

Кількісну характеристику ступеня операційного ризику запропоновано визначати як імовірність настання даного виду ризику, тобто імовірність ( $p_{OR}(H1)$ ) виникнення операційного ризику (подія  $H1$ ) за умови наявності інформації  $OR = (OR_1, OR_2, OR_3, OR_4, OR_5, OR_6, OR_7)$  в розрізі 7-х інцидентів, де  $OR_k, k=1 \div 7$  приймають значення 0, якщо відповідний норматив виконується (імовірність виникнення відповідного фактору ризику знаходиться у допустимих межах), і 1 – у супротивному випадку. Основою визначення складових  $OR = (OR_1, OR_2, OR_3, OR_4, OR_5, OR_6, OR_7)$  виступають імовірності ( $p_k(H1j)$ ) виникнення  $j$ -го інциденту операційного ризику (подія  $H1j$ ) за умови наявності інформації  $K = (K_1, K_2, \dots, K_n)$ , де  $K_k, k=1 \div n$  приймають значення 0, якщо відповідний норматив виконується, і 1 – у супротивному випадку.

Розглянемо послідовність визначення імовірності ( $p_{OR}(H1)$ ) виникнення операційного ризику (подія  $H1$ ) за умови наявності інформації  $OR = (OR_1, OR_2, OR_3, OR_4, OR_5, OR_6, OR_7)$ .

На основі отриманих бінарних показників трьох груп для кожного  $j$ -го інциденту ризику відповідно до формули Байєса (основа імовірнісного підходу), визначимо імовірність ( $p_k(H1j)$ ) виникнення  $j$ -го інциденту операційного ризику (подія  $H1j$ ) за умови наявності інформації  $K = (K_1, K_2, \dots, K_n)$  наступним чином (формули 1.8-1.9):

$$p_K(H1j) = \frac{1}{1 + e^{\{\lambda_0 + L\}}}, \quad (1.8)$$

$$L = \sum_{i=1}^n \lambda_i NKbin_{ij},$$

$$\lambda_{ij} = \ln \left( \frac{b_{ij}(1-g_{ij})}{g_{ij}(1-b_{ij})} \right), i=1, \dots, n, \quad (1.9)$$

$$\lambda_{0j} = \ln \left( \frac{p(H2j)}{p(H1j)} \right) + \sum_{i=1}^n \ln \left( \frac{1-b_{ij}}{1-g_{ij}} \right),$$

де  $p_K(H1j)$  – імовірність виникнення  $j$ -го інциденту операційного ризику за умови наявності інформації  $K = (K_1, K_2, \dots, K_n)$ ;

$L$  – інтегральний показник (зважена сума) бінарних характеристик  $NKbin_{ij}$  (наявна інформація про стан банку виходячи зі значень аналітичних показників);

$P(H1j)$  – імовірність настання гіпотези  $H1j$ ;

$H1j$  – висунута гіпотеза, що виникне  $j$ -й інциденту операційного ризику;

$P(H2j)$  – імовірність настання протилежної гіпотези;

$NK = \{NKbin_{ij}\}$  – бінарна компонента множини характеристик діяльності банку;

$b_{ij}$  – імовірність події  $NK = \{NKbin_{ij}\}$  для банку у розрізі  $j$ -го інциденту операційного ризику:

$$b_{ij} = 1 - g_{ij}; \quad (1.10)$$

$g_{ij}$  – імовірність протилежної події:

$$g_{ij} = \frac{\sum_i NKbin_{ij}}{n}. \quad (1.11)$$

Після визначення  $b_{ij}$  - імовірність події  $NKbin_{ij}=0$ , та  $g_{ij}$  - імовірність події  $NKbin_{ij}=1$  для кожного інциденту операційного ризику за всіма  $n$  показниками розрахуємо параметри  $\lambda_{ij}$  та  $\lambda_{0j}$  за формулами (формула 1.9), після чого визначимо значення  $L$  - інтегрального показника (зваженої суми) бінарних характеристик  $NK = \{NKbin_{ij}\}$  і підставимо в загальну формулу (формула 1.8), що показує величину оцінки ризику.

На основі отриманої імовірнісної (кількісної) оцінки ОР ( $p_K(H1j)$ ) за кожним  $j$ -м інцидентом визначається якісна характеристика рівня ризику:

– якщо  $0 \leq p_K(H1j) < fsr\{\min_s\{p_B(H1)_s\} \div fsr\{p_B(H1)_s\}\}$ , нормальний рівень ризику (де  $fsr\{\}$  - середнє значення зазначених показників за сукупністю  $s$  банків);

– якщо  $fsr\{\min_s\{p_B(H1)_s\} \div fsr\{p_B(H1)_s\}\} \leq p_K(H1j) < fsr\{p_B(H1)_s\}$ , підвищений рівень ризику;

– якщо  $fsr\{p_B(H1)_s\} \leq p_K(H1j) < fsr\left\{fsr\{p_B(H1)_s\} \div \max_s\{p_B(H1)_s\}\right\}$ , високий рівень ризику;

– якщо  $fsr\left\{fsr\{p_B(H1)_s\} \div \max_s\{p_B(H1)_s\}\right\} \leq p_K(H1j) \leq 1$ , критичний рівень ризику.

Використовуючи проведені вище розрахунки, визначимо алгоритм знаходження кількісної оцінки ступеня операційного ризику банку як імовірності виникнення операційного ризику за умови наявності інформації  $B = (p_K(H11), p_K(H12), p_K(H13), p_K(H14), p_K(H15), p_K(H16), p_K(H17))$ , яка визначається на основі аналітичних показників характеристики діяльності відповідного банку  $K = (K_1, K_2, \dots, K_n)$ :

а) розрахунок ймовірностей  $p_K(H1j)$  виникнення  $j$ -го інциденту операційного ризику за умови наявності інформації  $K = (K_1, K_2, \dots, K_n)$ ;

б) знаходження питомої ваги кожного з інцидентів у загальній структурі операційного ризику.  $S(p_K(H1j)) = \frac{p_K(H1j)}{\sum_{j=1}^7 p_K(H1j)} \times 100\%$ ;

в) визначення гранично допустимого коридору імовірнісної (кількісної) оцінки операційного ризику за кожним  $j$ -м інцидентом -  $0 \leq p_K(H1j) < 0,3$ , що визначає нормальний рівень ризику;

г) перехід від імовірнісних показників  $p_K(H1j)$  до бінарних показників  $NKbin_j$  за  $j$  інцидентами операційного ризику:  $NKbin_j$  приймає значення «1» у випадку влучення показника  $p_K(H1j)$  у гранично допустимі межі або значення «0» в іншому випадку;

г) визначення  $g_j$  - імовірності події  $NKbin_j = 1$  ( $g_{ij} = \frac{\sum_i NKbin_{ij}}{n}$ ) та  $b_j$  - імовірності події  $NKbin_{ij} = 0$  ( $b_{ij} = 1 - g_{ij}$ ) за  $j$  інцидентами операційного ризику;

д) розрахунок імовірності виникнення операційного ризику;

е) ідентифікація якісної оцінки рівня операційного ризику банку на основі визначеної кількісної оцінки його ступеня;

На основі отриманих бінарних показників  $NKbin_j$  за  $j$  інцидентами ризику відповідно до формули Байєса, яка є основою імовірнісного підходу, визначимо імовірність ( $p_B(H1)$ ) виникнення операційного ризику (подія  $H1$ ) за умови наявності інформації наступним чином (формула 1.12):

$$p_B(H1) = \frac{1}{1 + e^{\{\lambda_0 + L\}}}, \quad (1.12)$$

$$L = \sum_{j=1}^7 \lambda_j NKbin_j,$$

$$\lambda_j = \ln \left( \frac{b_j(1-g_j)}{g_j(1-b_j)} \right), j = 1, \dots, 7, \quad (1.13)$$

$$\lambda_{0j} = \ln \left( \frac{p(H2)}{p(H1)} \right) + \sum_{j=1}^7 \ln \left( \frac{1-b_j}{1-g_j} \right),$$

де  $p_B(H1)$  - імовірність виникнення операційного ризику за умови наявності інформації  $B=(p_K(H1), p_K(H2), p_K(H3), p_K(H4), p_K(H5), p_K(H6), p_K(H7))$ ;

$L$  - інтегральний показник (зважена сума) бінарних характеристик  $NKbin_j$ .

Логічним продовженням комплексу вище проведених розрахунків виступає реалізація сьомого етапу загальної моделі визначення рівня операційного ризику банківської установи, сутність якого полягає в надання якісної характеристики рівня даного ризику.

Так, на основі отриманої імовірнісної (кількісної) оцінки операційного ризику ( $p_B(H1)$ ) визначається якісна характеристика рівня ризику:

– якщо  $0 \leq p_B(H1) < fsr\{\min\{p_B(H1)_s\} \div fsr\{p_B(H1)_s\}\}$ , нормальний рівень ризику (де  $fsr\{\}$  - середнє значення зазначених показників за сукупністю  $s$  банків);

– якщо  $fsr\{\min\{p_B(H1)_s\} \div fsr\{p_B(H1)_s\}\} \leq p_B(H1) < fsr\{p_B(H1)_s\}$ , підвищений;

– якщо  $fsr\{p_B(H1)_s\} \leq p_B(H1) < fsr\left\{fsr\{p_B(H1)_s\} \div \max\{p_B(H1)_s\}\right\}$ , високий;

– якщо  $fsr\left\{fsr\{p_B(H1)_s\} \div \max\{p_B(H1)_s\}\right\} \leq p_B(H1) \leq 1$ , критичний рівень.

За результатами моделювання керівництво може обрати будь-яку зі стратегій управління операційним ризиком.

## 2 РОЗРОБКА ПРОГРАМНОГО ДОДАТКУ ДЛЯ РЕАЛІЗАЦІЇ МОДЕЛІ

При виборі мови програмування необхідно керуватися вимогами та підходами обраної технології, а також сучасними тенденціями розробки додатків на мовах програмування прикладного рівня. Тому для розробки програмного додатку було обрано мову програмування C #.

C # – об'єктно-орієнтована мова програмування, розроблена у 1998-2001 роках групою інженерів під керівництвом Андерса Гейлсберга в компанії Microsoft як основна мова розробки додатків для платформи Microsoft.NET. Компілятор з C # входить в стандартну установку самої .NET, тому програми на ньому можна створювати і компілювати навіть без інструментальних засобів. Мова має статичну типізацію, підтримує поліморфізм, перевантаження операторів (в тому числі операторів явного і неявного приведення типу), делегати, атрибути, події, властивості, узагальнені типи і методи, ітератори, анонімні функції з підтримкою замикань, виключення, коментарі у форматі XML.

Для поєднання обраної технології та мови програмування було визначено спільне середовище розробки програмних засобів прикладного рівня, а саме компоненту спеціалізованого засобу розробки орієнтованого на створення додатків баз даних Visual Studio фірми Microsoft – Visual C#.

Для створення програмного додатка розроблено блок-схему алгоритму реалізації моделі оцінки операційного ризику в комерційних банках (рисунок Б.1 додатку Б). Програмний код автоматизованого рішення представлено в додатку В. Початкове вікно, яке відкривається при вході в розроблений програмний додаток, показано на рисунку 2.1.

Меню програми було реалізовано за допомогою об'єкту C# MenuStrip і містить: головне меню; додаткові опції; довідку. Форма з трьома вкладками реалізована за допомогою об'єкта C# TableAdapter і містить: фінансові показники; бінарні характеристики фінансових показників; результати; проміжні розрахунки (рис. 2.1).

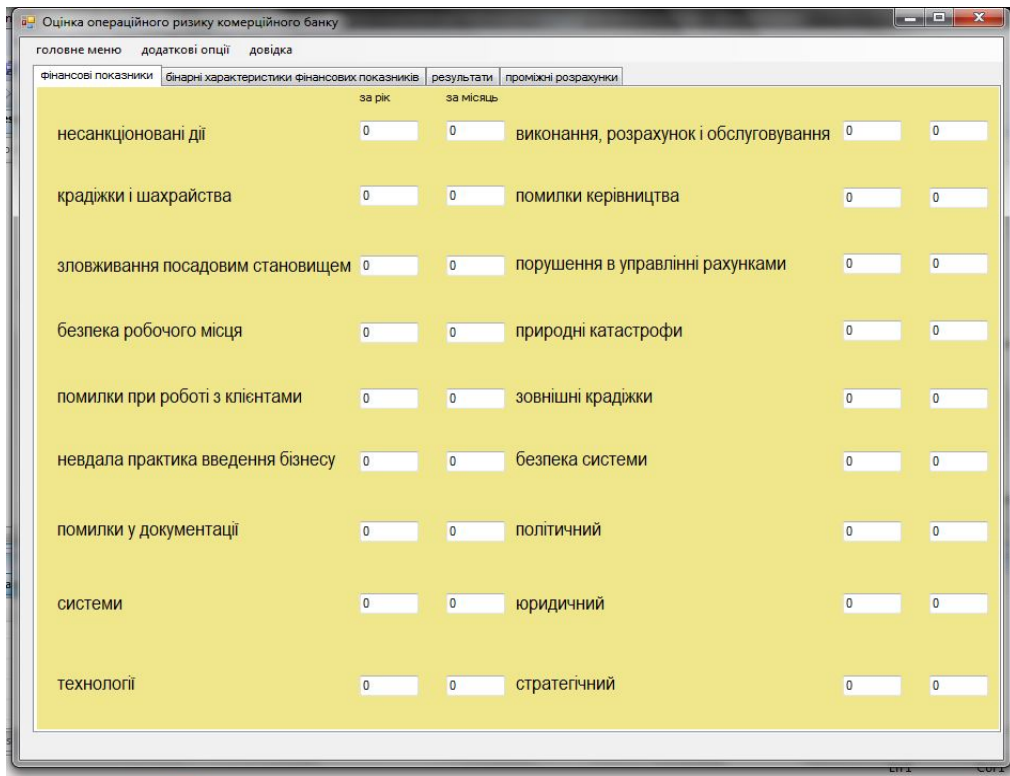


Рисунок 2.1 – Головне вікно програми

На рисунках 2.2 – 2.4 представлено складові кожного з пункту меню.

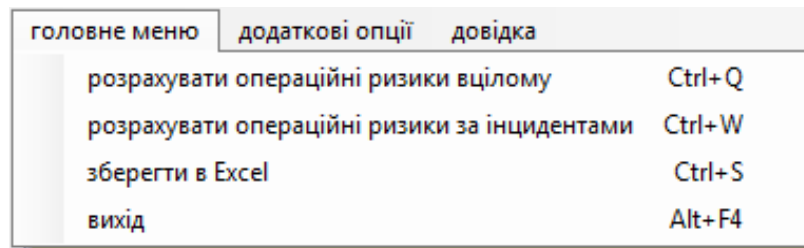


Рисунок 2.2 – Складові головного меню програми

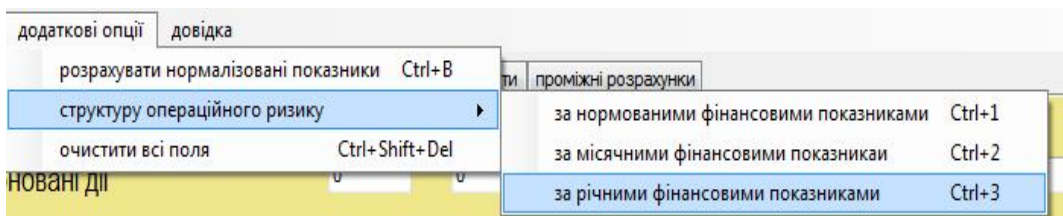


Рисунок 2.3 – Складові додаткових опцій програми

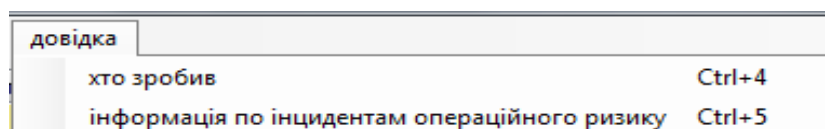


Рисунок 2.4 – Складові довідки програми



На рис. 2.5 – 2.6 представлено форми для заповнення вхідними даними.

	за рік	за місяць			
несанкціоновані дії	220	12	виконання, розрахунків і обслуговування	206	13
крадіжки і шахрайства	460	40	помилки керівництва	7	1
зловживання посадовим становищем	60	4	порушення в управлінні рахунками	23	4
безпека робочого місця	6	1	природні катастрофи	4	1
помилки при роботі з клієнтами	912	70	зовнішні крадіжки	9	2
невідала практика введення бізнесу	44	52	безпека системи	74	5
помилки у документації	6	2	політичний	7	1
системи	62	7	юридичний	9	2
технології	7	2	стратегічний	5	1

Рисунок 2.5 – Форма для заповнення фінансовими показниками (на прикладі «ОТР-банку»)

фінансові показники		бінарні характеристики фінансових показників						
		результати						
		проміжні розрахунки						
назва інциденту	інцидент 1-го роду	інцидент 2-го роду	інцидент 3-го роду	інцидент 4-го роду	інцидент 5-го роду	інцидент 6-го роду	інцидент 7-го роду	
▶ несанкціоновані...	1	0	1	0	1	1	1	
крадіжки і шахр...	1	1	1	1	1	1	1	
зловживання по...	1	0	1	0	1	1	0	
безпека робочо...	0	1	0	1	0	1	1	
помилки при ро...	1	0	1	1	1	0	1	
невідала практи...	0	1	1	1	1	1	1	
помилки у доку...	1	1	0	1	1	0	0	
системи	0	0	0	1	1	0	0	
технології	0	0	0	1	0	0	0	
виконання, роз...	0	0	1	1	1	0	0	
помилки керівн...	1	1	1	1	1	0	0	
порушення в уп...	1	1	1	1	1	0	1	
природні катаст...	0	0	0	0	1	1	1	
зовнішні крадіжки	0	0	0	1	1	1	1	
безпека системи	0	1	1	1	1	1	1	
політичний	0	0	0	0	0	0	1	
юридичний	0	0	1	0	1	0	1	
стратегічний	0	1	0	1	0	0	0	
*								

Рисунок 2.6 – Форма для заповнення бінарними характеристиками фінансових показників за кожним інцидентом

В програмному додатку існує дві можливості виводу результатів:

– результат оцінки операційного ризику по комерційному банку в цілому, для чого слід зайти в головне меню та обрати вкладку розрахувати операційні ризики в цілому (рисунок 2.7);

– результат оцінки кожного інциденту операційного ризику, для чого слід зайти в головне меню та обрати вкладку розрахувати операційні ризики за інцидентами (рисунок 2.8).

фінансові показники		бінарні характеристики фінансових показників		результати
	Інцидент ризику	Вірогідність настання		
▶*	Ризики операцій...	0,7		

Рисунок 2.7 – Результат розрахунку операційного ризику в цілому по банку

фінансові показники		бінарні характеристики фінансових показників		результати
	Інцидент ризику	Вірогідність настання		
	перший інцидент	0,61		
	другий інцидент	0,56		
	третій інцидент	0,44		
	четвертий інциде...	0,28		
	п'ятий інцидент	0,22		
	шостий інцидент	0,56		
▶*	сьомий інцидент	0,39		

Рисунок 2.8 – Результат розрахунку ймовірностей за кожним ОР

Було розроблено і інші допоміжні складові програми, наприклад, довідка по інцидентам (рисунок 2.9).

Перший інцидент - внутрішні шахрайства  
 Другий інцидент - підбір кадрів та умови праці  
 Третій інцидент - клієнти, продукти і практика бізнесу  
 Четвертий інцидент - збої в роботі і помилки систем  
 П'ятий інцидент - проведення операцій, а також керування процесами, пов'язаними з ними  
 Шостий інцидент - пошкодження активів  
 Сьомий інцидент - зовнішнє шахрайство

Рисунок 2.9 – Довідка по інцидентам операційного ризику.

Ще однією опцією програми є можливість розраховувати фінансові витрати за кожним інцидентом та всіма випадками ОР (рисунок 2.10)

показник ризику	1-й інцидент	2-й інцидент	3-й інцидент	4-й інцидент	5-й інцидент	6-й інцидент	7-й інцидент
несанкціоновані...	2,988	0,768	0,3	2,856	1,596	1,8	1,68
крадіжки і шахр...	114,54	29,44	11,5	109,48	61,18	69	64,4
зловживання по...	14,94	3,84	1,5	14,28	7,98	9	8,4
безпека робочо...	1,494	0,384	0,15	1,428	0,798	0,9	0,84
помилки при ро...	227,088	58,368	22,8	217,056	121,296	136,8	127,68
недала практи...	10,956	2,816	1,1	10,472	5,852	6,6	6,16
помилки у доку...	1,494	0,384	0,15	1,428	0,798	0,9	0,84
системи	15,438	3,968	1,55	14,756	8,246	9,3	8,68
технології	1,743	0,448	0,175	1,666	0,931	1,05	0,98
безпека системи	51,294	13,184	5,15	49,028	27,398	30,9	28,84
зовнішні крадіжки	1,743	0,448	0,175	1,666	0,931	1,05	0,98
природні катаст...	5,727	1,472	0,575	5,474	3,059	3,45	3,22
порушення в уп...	0,996	0,256	0,1	0,952	0,952	0,6	0,56
помилки керівн...	2,241	0,576	0,225	2,142	1,197	1,35	1,26
виконання, роз...	18,426	4,736	1,85	17,612	9,842	11,1	10,36
стратегічний	1,743	0,448	0,175	1,666	0,931	1,05	0,98
юридичний	2,241	0,576	0,225	2,142	1,197	1,35	1,26
політичний	1,245	0,32	0,125	1,19	0,665	0,75	0,7

Рисунок 2.10 – Відображення витрат за ризиками

Додатковою опцією для аналізу ОР є можливість будувати графіки в цілому за ризиком і за кожним інцидентом (рис. 2.11 – 2.13). Даний модуль є автономним, тому можна завантажувати дані не тільки з основної програми, а й з інших ресурсів.

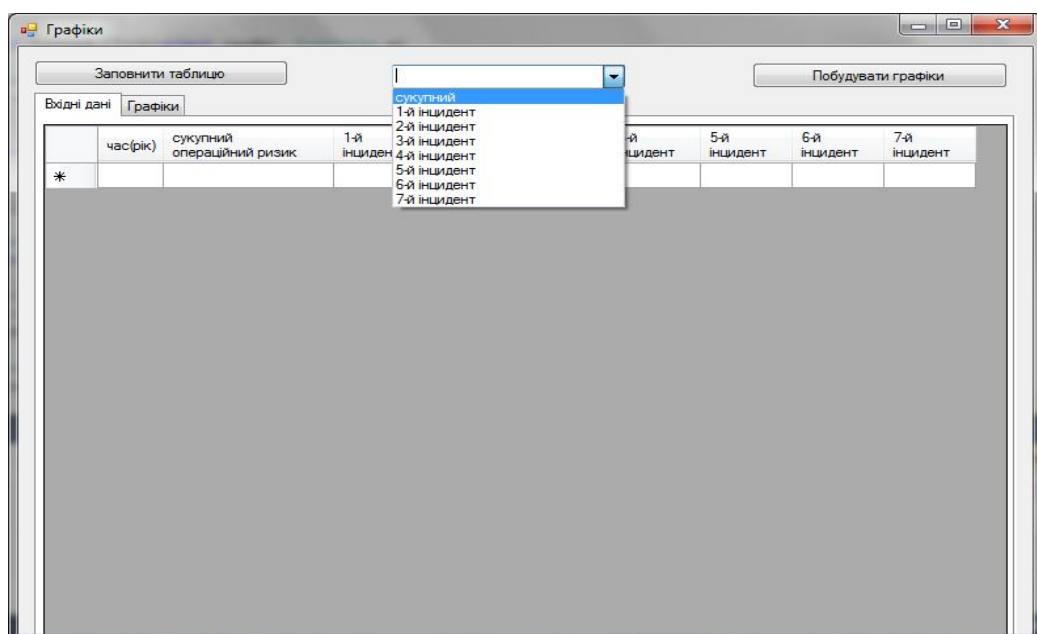


Рисунок 2.11 – Основне вікно допоміжної форми

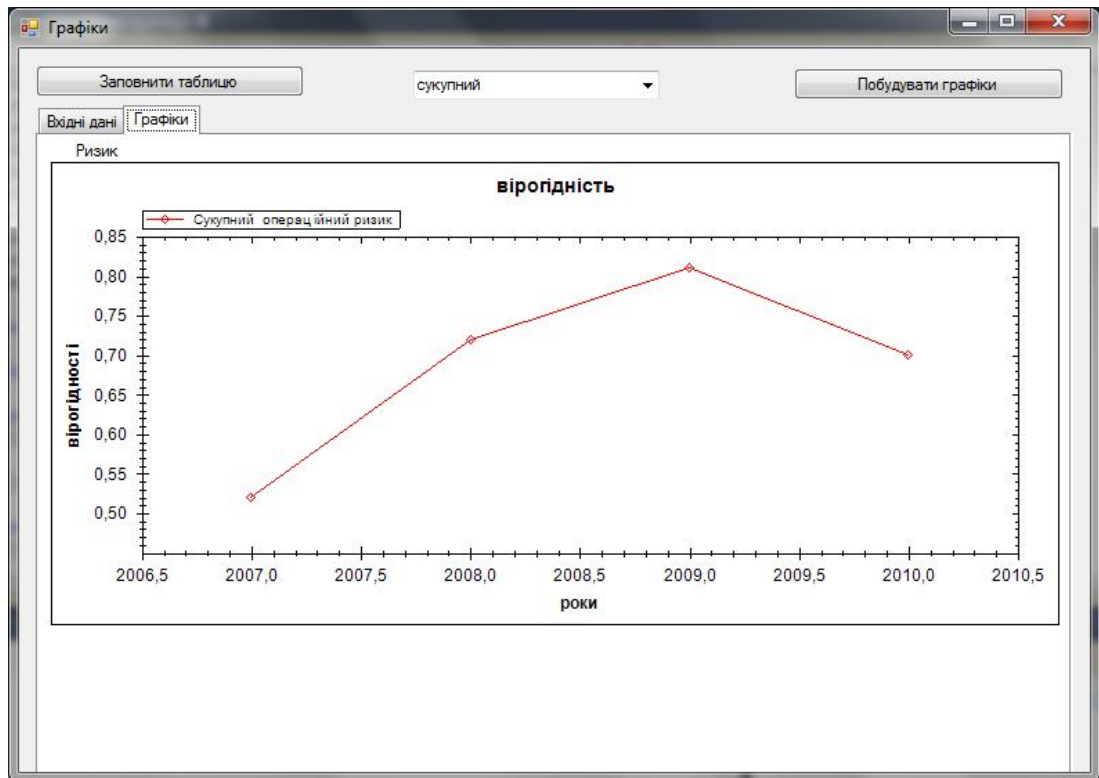


Рисунок 2.12 – Графічне зображення сукупного ОР за період 2007 – 2010 р.р.

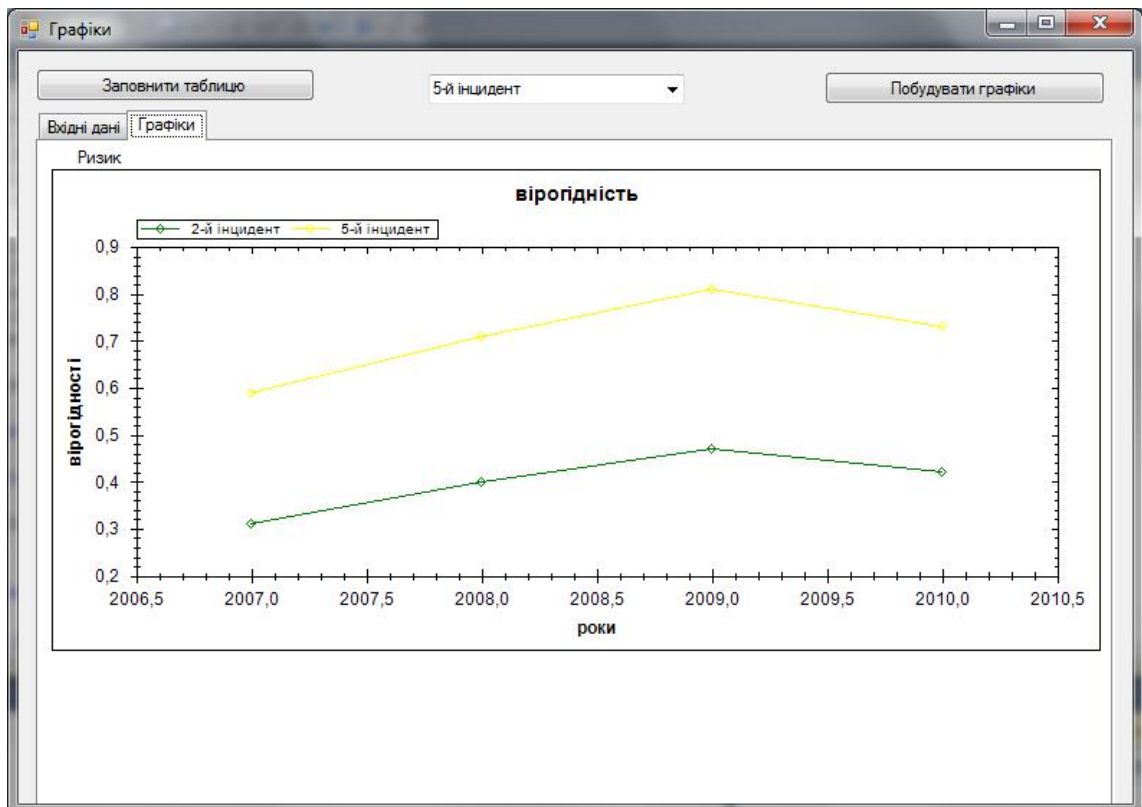


Рисунок 2.13 – ОР за різними інцидентами за 2007 – 2010 р.

Програмний додаток автоматизує розрахунки операційних ризиків і відповідає поставленим завданням моделі.

### 3 АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ ТА ПЕРЕВІРКА МОДЕЛІ НА ЧУТЛИВІСТЬ

Проведемо розрахунки за моделлю операційних ризиків за період з 2007 по 2010 роки (табл. 3.1).

Таблиця 3.1 – Оцінка операційного ризику «ОТР-банку» за період 2007-2010 роки.

Показник	Деталізація	Роки			
		2007	2008	2009	2010
Експрес-оцінка	Кількісна, од.	59	66	78	71
	Якісна	допустимий	високий	високий	високий
Імовірність виникнення інциденту операційного ризику, част. од.	Внутрішні шахрайства	0,44	0,44	0,52	0,35
	Підбір кадрів та умови праці	0,31	0,40	0,47	0,42
	Клієнти, продукти і практика бізнесу	0,58	0,59	0,71	0,48
	Збої в роботі і помилки систем	0,42	0,60	0,64	0,73
	Проведення операцій, а також керування процесами	0,59	0,71	0,81	0,73
	Пошкодження активів	0,29	0,35	0,33	0,42
	Зовнішнє шахрайство	0,51	0,49	0,50	0,46
Імовірність виникнення операційного ризику	Кількісна, част. од.	0,52	0,72	0,81	0,7
	Якісна	нормальний	високий	високий	високий
Зважена структура операційного ризику за інцидентами, %	Внутрішні шахрайства	13,98	12,21	13,07	9,71
	Підбір кадрів та умови праці	9,89	11,22	11,81	11,72
	Клієнти, продукти і практика бізнесу	18,35	16,45	17,84	13,26
	Збої в роботі і помилки систем	13,41	16,77	16,08	20,48
	Проведення операцій, а також керування процесами	18,83	19,92	20,35	20,39
	Пошкодження активів	9,26	9,72	8,29	11,72
	Зовнішнє шахрайство	16,28	13,70	12,56	12,73

Результати таблиці 3.1 свідчать про коливання практично всіх показників. Майже на всі вірогідності вплинула криза, підвищивши їх рівень. В після кризовий період вірогідності зменшуються. В 2010 році операційний ризик був на високому рівні, найбільше на нього вплинули інциденти, пов'язані зі збоями в роботі і помилками систем. На такому рівні впливав також інцидент, пов'язаний з проведенням операцій, а також керуванням процесами. На протязі трьох останніх років операційний ризик в «ОТР-банку» був на високому рівні, і більш за все це пояснювалось інцидентами, пов'язаними з збоями в роботі і помилками систем, та інцидентом пов'язаним з проведенням операцій, а також керуванням процесами. Протягом останніх років зменшились інциденти, пов'язані з внутрішніми шахрайствами, а також інцидент пов'язаний з клієнтами, продуктами і практикою бізнесу.

Модель є універсальною і за нею можна розраховувати рівень ризику в будь-якому комерційному банку. Розрахунки за даною моделлю може проводити ризик-менеджер, комітет по нагляду за банківськими ризиками. Розраховані результати за моделлю можуть допомогти майже будь-якому комітету банку, але найбільше правлінню банку, кредитному відділу.

Подібні моделі не набули впровадження в комерційних банках, тому не можливо перевірити її на адекватність. Використаємо перевірку моделі на її чутливість. Характеризуючи метод аналізу чутливості моделі, слід звернути увагу на те, що значення багатьох змінних величин, які використовуються в моделі, є величинами найбільш імовірними. Отже, аналіз чутливості дає можливість дати відповідь на запитання, який вплив на результат моделі матиме зміна у певних межах окремих параметрів.

Для якісного аналізу побудованої моделі необхідно провести розрахунок чутливості моделі. Для цього змінимо значення вхідних параметрів на 1%, 3%, 5%, 10%, 50%, 100% та проаналізуємо зміну результатів моделі (табл. 3.2-3.3).

Таблиця 3.2 – Перевірка чутливості фінансових результатів моделі

Зміна вхідних даних	Показник	Внутрішні шахрайства	Підбір кадрів та умови праці	Клієнти, продукти і практика бізнесу	Збої в роботі і помилки систем	Проведення операцій, а також керування процесами	Пошкодження активів	Зовнішнє шахрайство	Операційний ризик
Фактичне значення	розмір фінансових втрат, тис. грн.	2,99	0,77	0,30	3,10	3,03	1,80	1,68	13,68
100%	розмір фінансових втрат, тис. грн..	5,99	1,53	0,60	6,20	6,06	3,60	3,37	27,36
	темп приросту втрат, %	100	100	100	100	100	100	100	
50%	розмір фінансових втрат, тис. грн.	4,49	1,15	0,45	4,65	4,55	2,70	2,53	20,52
	темп приросту втрат, %	50	50	50	50	50	50	50	
10%	розмір фінансових втрат, тис. грн.	3,29	0,84	0,33	3,41	3,34	1,98	1,85	15,05
	темп приросту втрат, %	10	10	10	10	10	10	10	
5%	розмір фінансових втрат, тис. грн.	3,14	0,80	0,31	3,26	3,18	1,89	1,77	14,36
	темп приросту втрат, %	5	5	5	5	5	5	5	
3%	розмір фінансових втрат, тис. грн.	3,08	0,79	0,31	3,19	3,12	1,86	1,74	14,09
	темп приросту втрат, %	3	3	3	3	3	3	3	
1%	розмір фінансових втрат, тис. грн.	3,02	0,77	0,30	3,13	3,06	1,82	1,70	13,82
	темп приросту втрат, %	1	1	1	1	1	1	1	

З наведених результатів можна говорити, що відсоток зміни результату моделі відповідає відсотку зміни початкових даних, тому модель є не чутливою до зміни початкових даних.

Таблиця 3.3 – Перевірка чутливості імовірнісних результатів моделі

Зміна вхідних даних	Показник	Внутрішні шахрайства	Підбір кадрів та умови праці	Клієнти, продукти і практика бізнесу	Збої в роботі і помилки систем	Проведення операцій, а також керування процесами	Пошкодження активів	Зовнішні шахрайство
Фактичне значення	ймовірність ризику	0,348	0,420	0,475	0,734	0,731	0,420	0,456
100%	ймовірність ризику	0,55	0,60	0,58	0,70	0,71	0,57	0,65
	темп приросту ймовірності, %	56,82	43,10	21,31	-4,93	-2,60	35,39	42,82
50%	ймовірність ризику	0,47	0,52	0,49	0,74	0,75	0,49	0,56
	темп приросту ймовірності,%	33,93	23,13	4,06	0,78	2,22	16,02	23,56
10%	ймовірність ризику	0,362	0,405	0,491	0,755	0,710	0,435	0,514
	темп приросту ймовірності,%	4,07	-3,6 2	3,38	2,84	-2,8 4	3,62	12,67
5%	ймовірність ризику	0,391	0,435	0,411	0,712	0,751	0,466	0,481
	темп приросту ймовірності, %	12,32	3,51	-13,52	-2,96	2,72	10,98	5,34
3%	ймовірність ризику	0,396	0,440	0,416	0,719	0,758	0,412	0,486
	темп приросту ймовірності, %	13,74	4,74	-12,46	-2,03	3,69	-2,04	6,52
1%	ймовірність ризику	0,34	0,44	0,45	0,73	0,76	0,42	0,44
	темп приросту ймовірності, %	-2,28	3,59	-5,08	-1,10	4,66	-0,84	-3,25

З наведених результатів можна говорити про вплив зміни початкових даних на ймовірнісні характеристики. Цей вплив не є однорідним, хоча в більшості випадків зміна початкових даних була більше, ніж зміна ймовірності. Коливання ймовірності можна пояснити обернено експоненційним зв'язком між нею та вхідними даними. Також слід зауважити, що при збільшенні початкових результатів на більший відсоток, ймовірність збільшуються на менший, тобто зменшення чутливості моделі.

Перевірка чутливості модель показала, що побудована модель в цілому є не чутлива до змін початкових даних.



#### 4 ЕФЕКТ ВІД ВПРОВАДЖЕННЯ ЗАПРОПОНОВАНОЇ МОДЕЛІ ТА ПРОГРАМНОГО ДОДАТКУ

Оцінка ефективності є важливим показником для використання обраної моделі. Якщо впровадження системи є більш затратним ніж ефекти від впровадження, то необхідність в таких змінах автоматично відпадає. В розвинених країнах вже досить давно зрозуміли необхідність інновацій. І розуміють, що ефект може одразу ж не виявитись. В Україні звикли вкладати в проекти, в яких приріст капіталу за рік більш ніж 40%.

Оцінка ефективності для даної моделі є досить важкою задачею, і банки які не мають якогось фонду для зменшення операційного ризику не будуть її використовувати. В «Базелі II» досить велика увага приділяється оцінці операційних ризиків. Але в цьому документі всі оцінки виступають в ролі рекомендацій, а не обов'язків. В «Базелі III» рекомендації відносно операційних ризиків дещо змінюються, і починають носити декларативний характер. Отже можна прогнозувати, що майже всі банки опиняться перед вибором методу оцінки операційних ризиків, і тоді вони вже будуть змушені шукати найбільш ефективну модель.

Щоб краще оцінити ефект від впровадженої моделі слід відповісти на наступні питання:

- де буде проявлятись ефект?;
- на скільки довго даний ефект буде приносити користь?;
- який характер наслідків?;
- як проявиться ефект?;
- коли почнеться ефект?;
- яким буде вплив позитивним, чи ні?.

Тепер послідовно будемо відповідати на поставлені питання, тим самим визначаючи ефект від впровадження системи. Розроблена модель та програмний додаток будуть прямо впливати на внутрішній стан комерційного банку. Це пояснюється тим, що завдяки методиці оцінюються

ризиками та банк може зменшити їх, удосконалюючи якусь складову. Але не слід забувати, що одним з інцидентів ОР є зовнішні шахрайства, тому можна говорити про вплив і на зовнішній стан комерційного банку та намагання зменшити ризики ззовні, хоча і в меншій мірі.

Ефективність моделі і додатку принесе свою користь через певний проміжок часу. Через декілька років ефект від їх впровадження буде проявлятися в зменшенні операційних ризиків. Ефект від впровадження моделі та програмного додатку буде довготривалим.

Характер наслідків є економічним, бо в комерційному банку будуть зменшуватися витрати на покриття інцидентів, що класифікуються як операційні. Але не можна говорити про однозначний характер наслідків від впровадження. Крім економічного характеру, це торкнеться ще технічного, соціального та психологічного характеру наслідків від впровадження системи. Якщо найбільш складна ситуація з операційними ризиками четвертого роду, тобто систем, технологій, то комерційний банк буде вкладати кошти в розвиток саме цих складових, тим самим підвищуючи технічний розвиток. Якщо буде проблема з внутрішніми шахрайствами найбільш сильна, то зміни в банку будуть направлені на підвищення етнічних стандартів, психолого-соціальних норм. А це вже торкнеться ще однієї сторони від впровадження проекту.

Всі зміни, пов'язані з моделлю і додатком, будуть носити прямий характер, бо впливають безпосередньо на становище банку. Можна знайти деякі риси і опосередкованого впливу, але вони набагато менш значущі. Модель і додаток почнуть приносити ефект тільки після їх впровадження. В цілому можна очікувати позитивний вплив на становище банку і прогнозувати покращення його фінансових показників.

Також слід якісно охарактеризувати ефекти, які будуть отримано при вирішенні окремо кожного інциденту операційного ризику комерційного банку (таблиця 4.1).

Таблиця 4.1 – Характеристика ефекту від вирішення інцидентів операційного ризику

Інцидент ризику, який буде вирішуватись	Характеристика ефекту			
	За місцем його виявлення	За часом протягом якого одержується	За характером наслідків	За рівнем впливу
Внутрішні шахрайства	внутрішні	довготривалий	соціальний, етнічний, економічний, психологічний	опосередкований
Підбір кадрів та умови праці	внутрішні	довготривалий	соціальний, економічний	опосередкований
Клієнти, продукти і практика бізнесу	внутрішні	довготривалий	соціальний, економічний, психологічний	Опосередкований
Збої в роботі і помилки систем	внутрішні/ зовнішній	тимчасовий	технічний, економічний	Прямий
Проведення операцій, а також керування процесами	внутрішні	довготривалий	економічний, технічний, соціальний	прямий
Пошкодження активів	внутрішні	тимчасовий	економічний, правовий	опосередкований
Зовнішнє шахрайство	зовнішній	довготривалий	економічний, правовий	опосередкований

Основні ефекти від впровадження моделі і програмного додатку для оцінки операційного ризику в комерційному банку наступні:

- зниження операційного ризику;
- зменшення втрат пов'язаних з ризиками;
- вивільнення додаткового капіталу;
- збільшення доходу за рахунок вивільненого капіталу;
- зменшення внутрішніх та зовнішніх крадіжок;
- покращення роботи в колективі;
- оновлення технічного та програмного забезпечення;
- покращення менеджменту комерційного банку;
- відповідність комерційного банку європейським нормам;
- збільшення довіри інвесторів та вкладників до банку;
- підвищення прозорості в роботі банку;

- покращення фінансового стану банку;
- інші ефекти.

Розподіл ефекту за інцидентами представлений на рисунку 4.1.



Рисунок 4.1 – Опис ефектів в розрізі кожного інциденту

В цілому ефект від запропонованих моделі і додатку виявиться в скороченні витрат на операційні ризики і вивільнені додаткових ресурсів комерційного банку.

## ВИСНОВКИ

В діяльності будь-якого суб'єкту економічних відношень виникають ситуації, пов'язані із різного роду ризиками, обумовленими внутрішніми та зовнішніми факторами. Особливо до ризиків схильні операції комерційних банків, оскільки мають під собою фінансову основу. Тому доцільно вивчати ризики, досліджувати фактори впливу на них. Одним з ефективних інструментів дослідження є моделювання та автоматизація, що дозволить ризик-менеджеру банку змодельовати ризик, визначити його рівень та виробити рекомендації щодо його зменшення за рахунок тих факторів, що впливають. Відповідно зменшення ризику обумовлює зменшення витрат банку та збільшення його прибутків.

В роботі, використовуючи моделі оцінки операційних ризиків, виведено модель для оцінки операційного ризику за «Базелем III». Основою моделі для визначення операційного ризику комерційного банку виступає комбінація нечітко-множинного та імовірнісного (Баєсівського) підходів. Для реалізації моделі було розроблено програмний додаток, мовою програмування для якого обрано C#. З використанням додатку проведено апробацію моделі для «ОТР-банку» за період 2007 – 2010 роки і отримано результати щодо операційного ризику в цілому та за окремими інцидентами. Запропоновану модель перевірено на чутливість, в результаті чого зроблено висновок щодо не чутливості моделі до зміни вхідних даних.

Основними результатами від впровадження моделі та програмного додатку є: зниження операційних ризиків та зменшення втрат від них; вивільнення додаткового капіталу та збільшення за рахунок цього доходу; зменшення внутрішніх та зовнішніх крадіжок; покращення менеджменту банку; відповідність комерційного банку європейським нормам; збільшення довіри інвесторів та вкладників до банку та інше.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Камінський, А. Ідентифікація, аналіз та управління операційними ризиками в українських банках [Текст] / А. Камінський, А. Кияк // Вісник НБУ. – 2005. – № 10. – С. 7–11.
2. Kuritzkes, A. Operational risk Capital: A Problem of Definition [Text] / A. Kuritzkes // The Journal of Risk Finance. – 2002. – P. 47–56.
3. The New Basel Capital Accord [Text] / Basel committee on Banking Supervision. – 2003. –18 p.
4. Principles for financial market infrastructures [Text] / Basel committee on Banking Supervision. – 2011. –140 p.
5. Дмитрова, О.С. Моделювання оцінки операційного ризику комерційного банку [Текст] : монографія / О.С. Дмитрова, К.Г. Гончарова, О.В. Меренкова та ін. ; за заг. ред. С.О. Дмитрова. – Суми : ДВНЗ “УАБС НБУ”, 2010. – 264 с. - ISBN 978-966-8958-66-3.
6. International Convergence of Capital Measurement and Capital Standards – Bank for International Settlements [Text] / Basel Committee on Banking Supervision. – 2004. – 239 p.
7. Водзянова, Н.К. Концептуальні підходи до моделювання та управління операційним ризиком комерційного банку [Текст] / Н.К. Водзянова // Збірник наукових праць : Моделювання та інформаційні системи в економіці. – 2010. – Вип.№81. – С.11.
8. Даньків, В.Й. Теоретичні основи управління операційними ризиками [Текст] / В.Й. Даньків // Науковий вісник Ужгородського університету. – (Серія “Економіка”). – 2009. – Вип. 27. – С. 158–162.
9. Стубайло, Т. Місце операційних ризиків у банках України [Текст] / Т. Стубайло // Українська наука: минуле, сучасне, майбутнє : збірник наукових праць Тернопільського національного економічного університету. – 2007. – Вип. № 12. – С. 297–304.

## Додаток А

## Таблиця відповідності фінансових показників визначеним інцидентам операційного ризику

Таблиця А.1 – Відповідність фінансових показників визначеним інцидентам операційного ризику

Показник	Інцидент ризику (бінарна характеристика)						
	Внутрішні шахрайства	Підбір кадрів та умови праці	Клієнти, продукти і практика бізнесу	Збої в роботі і помилки систем	Проведення операцій, а також керування процесами	Пошкодження активів	Зовнішнє шахрайство
Назва показника	Бінарна характеристика						

## Додаток Б

Алгоритм реалізації моделі визначення рівня операційного ризику  
комерційного банку

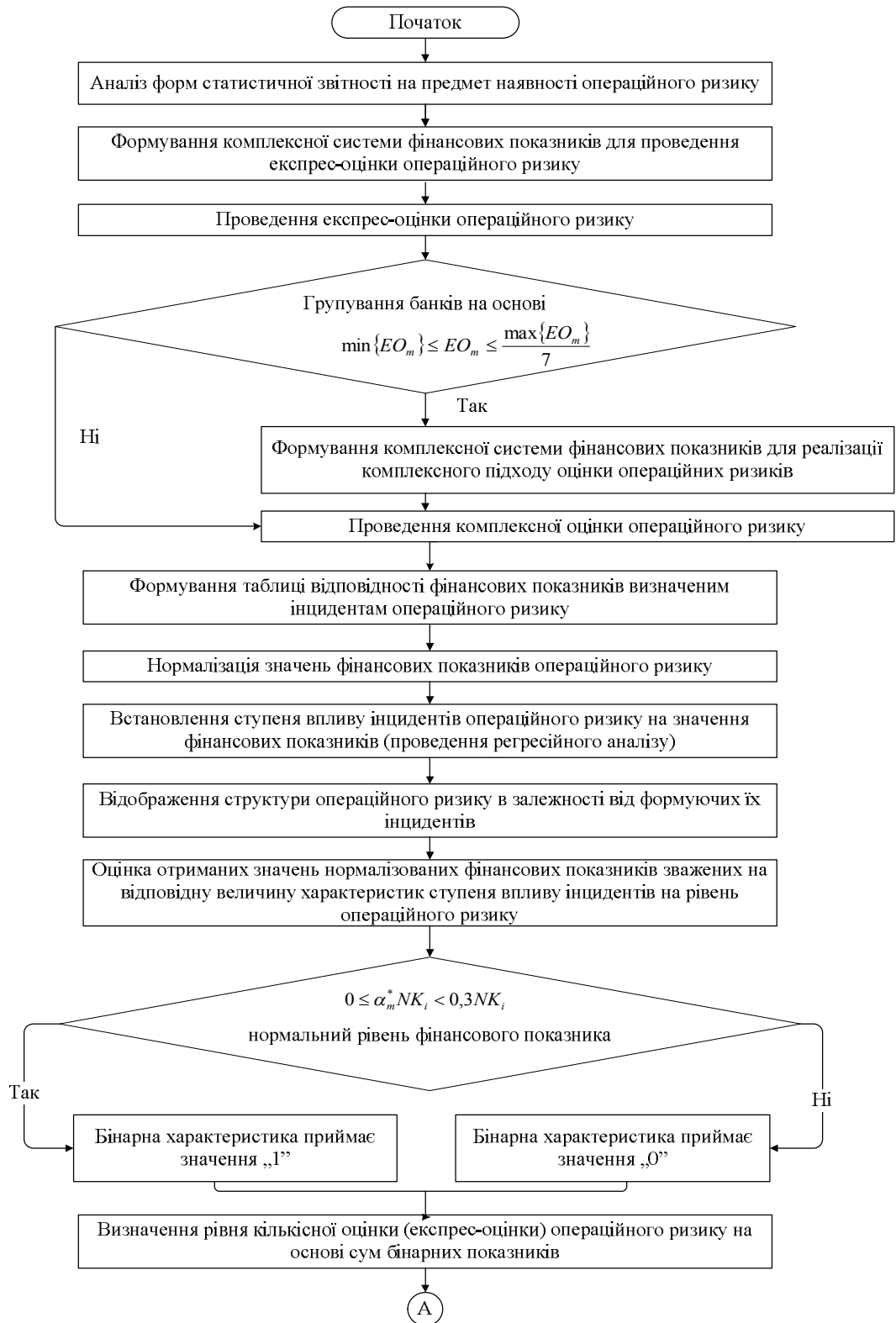
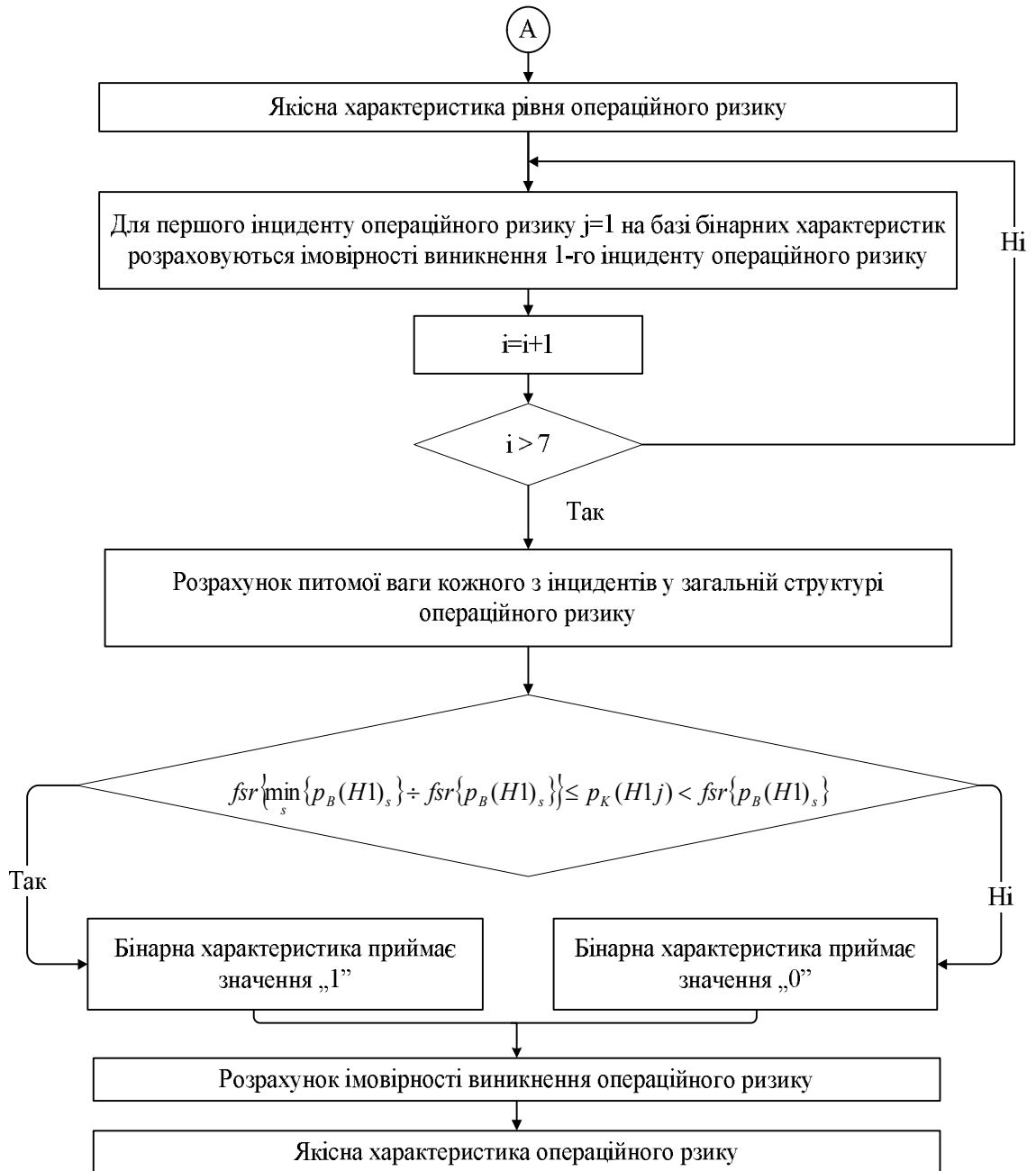


Рисунок Б.1 – Блок-схема алгоритму реалізації моделі оцінки операційного ризику в комерційних банках





Продовження рисунку Б.1



```

        dataGridView3.Rows.Add();
        dataGridView3.Rows.Add();
        dataGridView3.Rows.Add();
        dataGridView3.Rows.Add();
        dataGridView3.Rows.Add();
        dataGridView3.Rows.Add();
        dataGridView3.Rows.Add();
        dataGridView3.Rows[0].Cells[0].Value = label11.Text;
        dataGridView3.Rows[1].Cells[0].Value = label12.Text;
        dataGridView3.Rows[2].Cells[0].Value = label13.Text;
        dataGridView3.Rows[3].Cells[0].Value = label14.Text;
        dataGridView3.Rows[4].Cells[0].Value = label15.Text;
        dataGridView3.Rows[5].Cells[0].Value = label16.Text;
        dataGridView3.Rows[6].Cells[0].Value = label17.Text;
        dataGridView3.Rows[7].Cells[0].Value = label18.Text;
        dataGridView3.Rows[8].Cells[0].Value = label19.Text;
        dataGridView3.Rows[9].Cells[0].Value = label10.Text;
        dataGridView3.Rows[10].Cells[0].Value =
label111.Text;
        dataGridView3.Rows[11].Cells[0].Value =
label112.Text;
        dataGridView3.Rows[12].Cells[0].Value =
label113.Text;
        dataGridView3.Rows[13].Cells[0].Value =
label114.Text;
        dataGridView3.Rows[14].Cells[0].Value =
label115.Text;
        dataGridView3.Rows[15].Cells[0].Value =
label116.Text;
        dataGridView3.Rows[16].Cells[0].Value =
label117.Text;
        dataGridView3.Rows[17].Cells[0].Value =
label118.Text;
    }
    private void tabPage1_Click(object sender, EventArgs e)
    {
    }
    private void вихідToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        this.Close();
    }
    private void хтоЗробивToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        MessageBox.Show("Зробив Олексій Радовіченко");
    }
    private void dataGridView1_CellContentClick(object
sender, DataGridViewCellEventArgs e)
    {
    }
    private void Form1_Load(object sender, EventArgs e)
    {

```

// TODO: This line of code loads data into the 'diplomDataSet.binar' table. You can move, or remove it, as needed.

```

this.binarTableAdapter.Fill(this.diplomDataSet.binar);
    }
    private void button1_Click(object sender, EventArgs e)
    {
        // dataGridView1[1, 1].ToString;
    }
    private void tabPage3_Click(object sender, EventArgs e)
    {
    }
    private void
    розрахуватиОпераційніРизикиВціломуToolStripMenuItem_Click(object
    sender, EventArgs e)
    {
        double w1 = double.Parse(textBox2.Text) * 12 /
double.Parse(textBox1.Text);
        double w2 = double.Parse(textBox4.Text) * 12 /
double.Parse(textBox3.Text);
        double w3 = double.Parse(textBox6.Text) * 12 /
double.Parse(textBox5.Text);
        double w4 = double.Parse(textBox8.Text) * 12 /
double.Parse(textBox7.Text);
        double w5 = double.Parse(textBox10.Text) * 12 /
double.Parse(textBox9.Text);
        double w6 = double.Parse(textBox12.Text) * 12 /
double.Parse(textBox11.Text);
        double w7 = double.Parse(textBox14.Text) * 12 /
double.Parse(textBox13.Text);
        double w8 = double.Parse(textBox16.Text) * 12 /
double.Parse(textBox15.Text);
        double w9 = double.Parse(textBox18.Text) * 12 /
double.Parse(textBox17.Text);
        double w10 = double.Parse(textBox20.Text) * 12 /
double.Parse(textBox19.Text);
        double w11 = double.Parse(textBox22.Text) * 12 /
double.Parse(textBox21.Text);
        double w12 = double.Parse(textBox24.Text) * 12 /
double.Parse(textBox23.Text);
        double w13 = double.Parse(textBox26.Text) * 12 /
double.Parse(textBox25.Text);
        double w14 = double.Parse(textBox28.Text) * 12 /
double.Parse(textBox27.Text);
        double w15 = double.Parse(textBox30.Text) * 12 /
double.Parse(textBox29.Text);
        double w16 = double.Parse(textBox32.Text) * 12 /
double.Parse(textBox31.Text);
        double w17 = double.Parse(textBox34.Text) * 12 /
double.Parse(textBox33.Text);
        double w18 = double.Parse(textBox36.Text) * 12 /
double.Parse(textBox35.Text);
    }

```

```
double g1 = 0.249, g2 = 0.064, g3 = 0.025, g4 =
0.238, g5 = 0.133, g6 = 0.150, g7 = 0.140;
dataGridView3.Rows[0].Cells[1].Value = w1 * g1;
dataGridView3.Rows[1].Cells[1].Value = w2 * g1;
dataGridView3.Rows[2].Cells[1].Value = w3 * g1;
dataGridView3.Rows[3].Cells[1].Value = w4 * g1;
dataGridView3.Rows[4].Cells[1].Value = w5 * g1;
dataGridView3.Rows[5].Cells[1].Value = w6 * g1;
dataGridView3.Rows[6].Cells[1].Value = w7 * g1;
dataGridView3.Rows[7].Cells[1].Value = w8 * g1;
dataGridView3.Rows[8].Cells[1].Value = w9 * g1;
dataGridView3.Rows[9].Cells[1].Value = w10 * g1;
dataGridView3.Rows[10].Cells[1].Value = w11 * g1;
dataGridView3.Rows[11].Cells[1].Value = w12 * g1;
dataGridView3.Rows[12].Cells[1].Value = w13 * g1;
dataGridView3.Rows[13].Cells[1].Value = w14 * g1;
dataGridView3.Rows[14].Cells[1].Value = w15 * g1;
dataGridView3.Rows[15].Cells[1].Value = w16 * g1;
dataGridView3.Rows[16].Cells[1].Value = w17 * g1;
dataGridView3.Rows[17].Cells[1].Value = w18 * g1;
dataGridView3.Rows[0].Cells[2].Value = w1 * g2;
dataGridView3.Rows[1].Cells[2].Value = w2 * g2;
dataGridView3.Rows[2].Cells[2].Value = w3 * g2;
dataGridView3.Rows[3].Cells[2].Value = w4 * g2;
dataGridView3.Rows[4].Cells[2].Value = w5 * g2;
dataGridView3.Rows[5].Cells[2].Value = w6 * g2;
dataGridView3.Rows[6].Cells[2].Value = w7 * g2;
dataGridView3.Rows[7].Cells[2].Value = w8 * g2;
dataGridView3.Rows[8].Cells[2].Value = w9 * g2;
dataGridView3.Rows[9].Cells[2].Value = w10 * g2;
dataGridView3.Rows[10].Cells[2].Value = w11 * g2;
dataGridView3.Rows[11].Cells[2].Value = w12 * g2;
dataGridView3.Rows[12].Cells[2].Value = w13 * g2;
dataGridView3.Rows[13].Cells[2].Value = w14 * g2;
dataGridView3.Rows[14].Cells[2].Value = w15 * g2;
dataGridView3.Rows[15].Cells[2].Value = w16 * g2;
dataGridView3.Rows[16].Cells[2].Value = w17 * g2;
dataGridView3.Rows[17].Cells[2].Value = w18 * g2;
dataGridView3.Rows[0].Cells[3].Value = w1 * g3;
dataGridView3.Rows[1].Cells[3].Value = w2 * g3;
dataGridView3.Rows[2].Cells[3].Value = w3 * g3;
dataGridView3.Rows[3].Cells[3].Value = w4 * g3;
dataGridView3.Rows[4].Cells[3].Value = w5 * g3;
dataGridView3.Rows[5].Cells[3].Value = w6 * g3;
dataGridView3.Rows[6].Cells[3].Value = w7 * g3;
dataGridView3.Rows[7].Cells[3].Value = w8 * g3;
dataGridView3.Rows[8].Cells[3].Value = w9 * g3;
dataGridView3.Rows[9].Cells[3].Value = w10 * g3;
dataGridView3.Rows[10].Cells[3].Value = w11 * g3;
dataGridView3.Rows[11].Cells[3].Value = w12 * g3;
dataGridView3.Rows[12].Cells[3].Value = w13 * g3;
dataGridView3.Rows[13].Cells[3].Value = w14 * g3;
dataGridView3.Rows[14].Cells[3].Value = w15 * g3;
```

```
dataGridView3.Rows[15].Cells[3].Value = w16 * g3;
dataGridView3.Rows[16].Cells[3].Value = w17 * g3;
dataGridView3.Rows[17].Cells[3].Value = w18 * g3;
dataGridView3.Rows[0].Cells[4].Value = w1 * g4;
dataGridView3.Rows[1].Cells[4].Value = w2 * g4;
dataGridView3.Rows[2].Cells[4].Value = w3 * g4;
dataGridView3.Rows[3].Cells[4].Value = w4 * g4;
dataGridView3.Rows[4].Cells[4].Value = w5 * g4;
dataGridView3.Rows[5].Cells[4].Value = w6 * g4;
dataGridView3.Rows[6].Cells[4].Value = w7 * g4;
dataGridView3.Rows[7].Cells[4].Value = w8 * g4;
dataGridView3.Rows[8].Cells[4].Value = w9 * g4;
dataGridView3.Rows[9].Cells[4].Value = w10 * g4;
dataGridView3.Rows[10].Cells[4].Value = w11 * g4;
dataGridView3.Rows[11].Cells[4].Value = w12 * g4;
dataGridView3.Rows[12].Cells[4].Value = w13 * g4;
dataGridView3.Rows[13].Cells[4].Value = w14 * g4;
dataGridView3.Rows[14].Cells[4].Value = w15 * g4;
dataGridView3.Rows[15].Cells[4].Value = w16 * g4;
dataGridView3.Rows[16].Cells[4].Value = w17 * g4;
dataGridView3.Rows[17].Cells[4].Value = w18 * g4;
dataGridView3.Rows[0].Cells[5].Value = w1 * g5;
dataGridView3.Rows[1].Cells[5].Value = w2 * g5;
dataGridView3.Rows[2].Cells[5].Value = w3 * g5;
dataGridView3.Rows[3].Cells[5].Value = w4 * g5;
dataGridView3.Rows[4].Cells[5].Value = w5 * g5;
dataGridView3.Rows[5].Cells[5].Value = w6 * g5;
dataGridView3.Rows[6].Cells[5].Value = w7 * g5;
dataGridView3.Rows[7].Cells[5].Value = w8 * g5;
dataGridView3.Rows[8].Cells[5].Value = w9 * g5;
dataGridView3.Rows[9].Cells[5].Value = w10 * g5;
dataGridView3.Rows[10].Cells[5].Value = w11 * g5;
dataGridView3.Rows[11].Cells[5].Value = w12 * g5;
dataGridView3.Rows[12].Cells[5].Value = w13 * g4;
dataGridView3.Rows[13].Cells[5].Value = w14 * g5;
dataGridView3.Rows[14].Cells[5].Value = w15 * g5;
dataGridView3.Rows[15].Cells[5].Value = w16 * g5;
dataGridView3.Rows[16].Cells[5].Value = w17 * g5;
dataGridView3.Rows[17].Cells[5].Value = w18 * g5;
dataGridView3.Rows[0].Cells[6].Value = w1 * g6;
dataGridView3.Rows[1].Cells[6].Value = w2 * g6;
dataGridView3.Rows[2].Cells[6].Value = w3 * g6;
dataGridView3.Rows[3].Cells[6].Value = w4 * g6;
dataGridView3.Rows[4].Cells[6].Value = w5 * g6;
dataGridView3.Rows[5].Cells[6].Value = w6 * g6;
dataGridView3.Rows[6].Cells[6].Value = w7 * g6;
dataGridView3.Rows[7].Cells[6].Value = w8 * g6;
dataGridView3.Rows[8].Cells[6].Value = w9 * g6;
dataGridView3.Rows[9].Cells[6].Value = w10 * g6;
dataGridView3.Rows[10].Cells[6].Value = w11 * g6;
dataGridView3.Rows[11].Cells[6].Value = w12 * g6;
dataGridView3.Rows[12].Cells[6].Value = w13 * g6;
dataGridView3.Rows[13].Cells[6].Value = w14 * g6;
```

```

dataGridView3.Rows[14].Cells[6].Value = w15 * g6;
dataGridView3.Rows[15].Cells[6].Value = w16 * g6;
dataGridView3.Rows[16].Cells[6].Value = w17 * g6;
dataGridView3.Rows[17].Cells[6].Value = w18 * g6;
dataGridView3.Rows[0].Cells[7].Value = w1 * g7;
dataGridView3.Rows[1].Cells[7].Value = w2 * g7;
dataGridView3.Rows[2].Cells[7].Value = w3 * g7;
dataGridView3.Rows[3].Cells[7].Value = w4 * g7;
dataGridView3.Rows[4].Cells[7].Value = w5 * g7;
dataGridView3.Rows[5].Cells[7].Value = w6 * g7;
dataGridView3.Rows[6].Cells[7].Value = w7 * g7;
dataGridView3.Rows[7].Cells[7].Value = w8 * g7;
dataGridView3.Rows[8].Cells[7].Value = w9 * g7;
dataGridView3.Rows[9].Cells[7].Value = w10 * g7;
dataGridView3.Rows[10].Cells[7].Value = w11 * g7;
dataGridView3.Rows[11].Cells[7].Value = w12 * g7;
dataGridView3.Rows[12].Cells[7].Value = w13 * g7;
dataGridView3.Rows[13].Cells[7].Value = w14 * g7;
dataGridView3.Rows[14].Cells[7].Value = w15 * g7;
dataGridView3.Rows[15].Cells[7].Value = w16 * g7;
dataGridView3.Rows[16].Cells[7].Value = w17 * g7;
dataGridView3.Rows[17].Cells[7].Value = w18 * g7;
double s=0;
for (int i=1; i < dataGridView1.ColumnCount; i++)
{
    for (int j=0; j < dataGridView1.RowCount; j++)
    {
        s=s+
double.Parse(dataGridView1.Rows[j].Cells[i].Value.ToString);
    }
}

s=s/(dataGridView1.ColumnCount*dataGridView1.RowCount);
dataGridView2.Rows[0].Cells[0].Value = "Ризики
операційного банку в цілому";
dataGridView2.Rows[0].Cells[1].Value = s;
}
private void
розрахуватиОпераційніРизикиЗаІнцидентамиToolStripMenuItem_Click(
object sender, EventArgs e)
{
    double w1 = double.Parse(textBox2.Text) * 12 /
double.Parse(textBox1.Text);
    double w2 = double.Parse(textBox4.Text) * 12 /
double.Parse(textBox3.Text);
    double w3 = double.Parse(textBox6.Text) * 12 /
double.Parse(textBox5.Text);
    double w4 = double.Parse(textBox8.Text) * 12 /
double.Parse(textBox7.Text);
    double w5 = double.Parse(textBox10.Text) * 12 /
double.Parse(textBox9.Text);
    double w6 = double.Parse(textBox12.Text) * 12 /
double.Parse(textBox11.Text);
}

```

```

        double w7 = double.Parse(textBox14.Text) * 12 /
double.Parse(textBox13.Text);
        double w8 = double.Parse(textBox16.Text) * 12 /
double.Parse(textBox15.Text);
        double w9 = double.Parse(textBox18.Text) * 12 /
double.Parse(textBox17.Text);
        double w10 = double.Parse(textBox20.Text) * 12 /
double.Parse(textBox19.Text);
        double w11 = double.Parse(textBox22.Text) * 12 /
double.Parse(textBox21.Text);
        double w12 = double.Parse(textBox24.Text) * 12 /
double.Parse(textBox23.Text);
        double w13 = double.Parse(textBox26.Text) * 12 /
double.Parse(textBox25.Text);
        double w14 = double.Parse(textBox28.Text) * 12 /
double.Parse(textBox27.Text);
        double w15 = double.Parse(textBox30.Text) * 12 /
double.Parse(textBox29.Text);
        double w16 = double.Parse(textBox32.Text) * 12 /
double.Parse(textBox31.Text);
        double w17 = double.Parse(textBox34.Text) * 12 /
double.Parse(textBox33.Text);
        double w18 = double.Parse(textBox36.Text) * 12 /
double.Parse(textBox35.Text);
        double g1 = 0.249, g2 = 0.064, g3 = 0.025, g4 =
0.238, g5 = 0.133, g6 = 0.150, g7 = 0.140;
        dataGridView3.Rows[0].Cells[1].Value = w1 * g1;
        dataGridView3.Rows[1].Cells[1].Value = w2 * g1;
        dataGridView3.Rows[2].Cells[1].Value = w3 * g1;
        dataGridView3.Rows[3].Cells[1].Value = w4 * g1;
        dataGridView3.Rows[4].Cells[1].Value = w5 * g1;
        dataGridView3.Rows[5].Cells[1].Value = w6 * g1;
        dataGridView3.Rows[6].Cells[1].Value = w7 * g1;
        dataGridView3.Rows[7].Cells[1].Value = w8 * g1;
        dataGridView3.Rows[8].Cells[1].Value = w9 * g1;
        dataGridView3.Rows[9].Cells[1].Value = w10 * g1;
        dataGridView3.Rows[10].Cells[1].Value = w11 * g1;
        dataGridView3.Rows[11].Cells[1].Value = w12 * g1;
        dataGridView3.Rows[12].Cells[1].Value = w13 * g1;
        dataGridView3.Rows[13].Cells[1].Value = w14 * g1;
        dataGridView3.Rows[14].Cells[1].Value = w15 * g1;
        dataGridView3.Rows[15].Cells[1].Value = w16 * g1;
        dataGridView3.Rows[16].Cells[1].Value = w17 * g1;
        dataGridView3.Rows[17].Cells[1].Value = w18 * g1;
        dataGridView3.Rows[0].Cells[2].Value = w1 * g2;
        dataGridView3.Rows[1].Cells[2].Value = w2 * g2;
        dataGridView3.Rows[2].Cells[2].Value = w3 * g2;
        dataGridView3.Rows[3].Cells[2].Value = w4 * g2;
        dataGridView3.Rows[4].Cells[2].Value = w5 * g2;
        dataGridView3.Rows[5].Cells[2].Value = w6 * g2;
        dataGridView3.Rows[6].Cells[2].Value = w7 * g2;
        dataGridView3.Rows[7].Cells[2].Value = w8 * g2;
        dataGridView3.Rows[8].Cells[2].Value = w9 * g2;

```



```
dataGridView3.Rows[9].Cells[2].Value = w10 * g2;
dataGridView3.Rows[10].Cells[2].Value = w11 * g2;
dataGridView3.Rows[11].Cells[2].Value = w12 * g2;
dataGridView3.Rows[12].Cells[2].Value = w13 * g2;
dataGridView3.Rows[13].Cells[2].Value = w14 * g2;
dataGridView3.Rows[14].Cells[2].Value = w15 * g2;
dataGridView3.Rows[15].Cells[2].Value = w16 * g2;
dataGridView3.Rows[16].Cells[2].Value = w17 * g2;
dataGridView3.Rows[17].Cells[2].Value = w18 * g2;
dataGridView3.Rows[0].Cells[3].Value = w1 * g3;
dataGridView3.Rows[1].Cells[3].Value = w2 * g3;
dataGridView3.Rows[2].Cells[3].Value = w3 * g3;
dataGridView3.Rows[3].Cells[3].Value = w4 * g3;
dataGridView3.Rows[4].Cells[3].Value = w5 * g3;
dataGridView3.Rows[5].Cells[3].Value = w6 * g3;
dataGridView3.Rows[6].Cells[3].Value = w7 * g3;
dataGridView3.Rows[7].Cells[3].Value = w8 * g3;
dataGridView3.Rows[8].Cells[3].Value = w9 * g3;
dataGridView3.Rows[9].Cells[3].Value = w10 * g3;
dataGridView3.Rows[10].Cells[3].Value = w11 * g3;
dataGridView3.Rows[11].Cells[3].Value = w12 * g3;
dataGridView3.Rows[12].Cells[3].Value = w13 * g3;
dataGridView3.Rows[13].Cells[3].Value = w14 * g3;
dataGridView3.Rows[14].Cells[3].Value = w15 * g3;
dataGridView3.Rows[15].Cells[3].Value = w16 * g3;
dataGridView3.Rows[16].Cells[3].Value = w17 * g3;
dataGridView3.Rows[17].Cells[3].Value = w18 * g3;
dataGridView3.Rows[0].Cells[4].Value = w1 * g4;
dataGridView3.Rows[1].Cells[4].Value = w2 * g4;
dataGridView3.Rows[2].Cells[4].Value = w3 * g4;
dataGridView3.Rows[3].Cells[4].Value = w4 * g4;
dataGridView3.Rows[4].Cells[4].Value = w5 * g4;
dataGridView3.Rows[5].Cells[4].Value = w6 * g4;
dataGridView3.Rows[6].Cells[4].Value = w7 * g4;
dataGridView3.Rows[7].Cells[4].Value = w8 * g4;
dataGridView3.Rows[8].Cells[4].Value = w9 * g4;
dataGridView3.Rows[9].Cells[4].Value = w10 * g4;
dataGridView3.Rows[10].Cells[4].Value = w11 * g4;
dataGridView3.Rows[11].Cells[4].Value = w12 * g4;
dataGridView3.Rows[12].Cells[4].Value = w13 * g4;
dataGridView3.Rows[13].Cells[4].Value = w14 * g4;
dataGridView3.Rows[14].Cells[4].Value = w15 * g4;
dataGridView3.Rows[15].Cells[4].Value = w16 * g4;
dataGridView3.Rows[16].Cells[4].Value = w17 * g4;
dataGridView3.Rows[17].Cells[4].Value = w18 * g4;
dataGridView3.Rows[0].Cells[5].Value = w1 * g5;
dataGridView3.Rows[1].Cells[5].Value = w2 * g5;
dataGridView3.Rows[2].Cells[5].Value = w3 * g5;
dataGridView3.Rows[3].Cells[5].Value = w4 * g5;
dataGridView3.Rows[4].Cells[5].Value = w5 * g5;
dataGridView3.Rows[5].Cells[5].Value = w6 * g5;
dataGridView3.Rows[6].Cells[5].Value = w7 * g5;
dataGridView3.Rows[7].Cells[5].Value = w8 * g5;
```

```

dataGridView3.Rows[8].Cells[5].Value = w9 * g5;
dataGridView3.Rows[9].Cells[5].Value = w10 * g5;
dataGridView3.Rows[10].Cells[5].Value = w11 * g5;
dataGridView3.Rows[11].Cells[5].Value = w12 * g5;
dataGridView3.Rows[12].Cells[5].Value = w13 * g4;
dataGridView3.Rows[13].Cells[5].Value = w14 * g5;
dataGridView3.Rows[14].Cells[5].Value = w15 * g5;
dataGridView3.Rows[15].Cells[5].Value = w16 * g5;
dataGridView3.Rows[16].Cells[5].Value = w17 * g5;
dataGridView3.Rows[17].Cells[5].Value = w18 * g5;
dataGridView3.Rows[0].Cells[6].Value = w1 * g6;
dataGridView3.Rows[1].Cells[6].Value = w2 * g6;
dataGridView3.Rows[2].Cells[6].Value = w3 * g6;
dataGridView3.Rows[3].Cells[6].Value = w4 * g6;
dataGridView3.Rows[4].Cells[6].Value = w5 * g6;
dataGridView3.Rows[5].Cells[6].Value = w6 * g6;
dataGridView3.Rows[6].Cells[6].Value = w7 * g6;
dataGridView3.Rows[7].Cells[6].Value = w8 * g6;
dataGridView3.Rows[8].Cells[6].Value = w9 * g6;
dataGridView3.Rows[9].Cells[6].Value = w10 * g6;
dataGridView3.Rows[10].Cells[6].Value = w11 * g6;
dataGridView3.Rows[11].Cells[6].Value = w12 * g6;
dataGridView3.Rows[12].Cells[6].Value = w13 * g6;
dataGridView3.Rows[13].Cells[6].Value = w14 * g6;
dataGridView3.Rows[14].Cells[6].Value = w15 * g6;
dataGridView3.Rows[15].Cells[6].Value = w16 * g6;
dataGridView3.Rows[16].Cells[6].Value = w17 * g6;
dataGridView3.Rows[17].Cells[6].Value = w18 * g6;
dataGridView3.Rows[0].Cells[7].Value = w1 * g7;
dataGridView3.Rows[1].Cells[7].Value = w2 * g7;
dataGridView3.Rows[2].Cells[7].Value = w3 * g7;
dataGridView3.Rows[3].Cells[7].Value = w4 * g7;
dataGridView3.Rows[4].Cells[7].Value = w5 * g7;
dataGridView3.Rows[5].Cells[7].Value = w6 * g7;
dataGridView3.Rows[6].Cells[7].Value = w7 * g7;
dataGridView3.Rows[7].Cells[7].Value = w8 * g7;
dataGridView3.Rows[8].Cells[7].Value = w9 * g7;
dataGridView3.Rows[9].Cells[7].Value = w10 * g7;
dataGridView3.Rows[10].Cells[7].Value = w11 * g7;
dataGridView3.Rows[11].Cells[7].Value = w12 * g7;
dataGridView3.Rows[12].Cells[7].Value = w13 * g7;
dataGridView3.Rows[13].Cells[7].Value = w14 * g7;
dataGridView3.Rows[14].Cells[7].Value = w15 * g7;
dataGridView3.Rows[15].Cells[7].Value = w16 * g7;
dataGridView3.Rows[16].Cells[7].Value = w17 * g7;
dataGridView3.Rows[17].Cells[7].Value = w18 * g7;
dataGridView2.Rows[0].Cells[0].Value = "перший
інцидент";
dataGridView2.Rows[1].Cells[0].Value = "другий
інцидент";
dataGridView2.Rows[2].Cells[0].Value = "третій
інцидент";

```

```

        dataGridView2.Rows[3].Cells[0].Value = "четвертий
інцидент";
        dataGridView2.Rows[4].Cells[0].Value = "пятий
інцидент";
        dataGridView2.Rows[5].Cells[0].Value = "шостий
інцидент";
        dataGridView2.Rows[6].Cells[0].Value = "сьомий
інцидент";
        double s = 0;
        for (int i = 1; i < dataGridView1.ColumnCount; i++)
        {
            s=0;
            for (int j = 0; j < dataGridView1.RowCount; j++)
            {
                s = s +
double.Parse(dataGridView1.Rows[j].Cells[i].Value.ToString);
            }
            s=s/dataGridView1.RowCount;
            dataGridView2.Rows[i].Cells[0].Value = s;
        }
    }
    private void
інформаціяПоІнцидентамОпераційногоРизикуToolStripMenuItem_Click(
object sender, EventArgs e)
    {
        Form frm= new Form2();
        frm.Show();
    }
    private void
розрахуватиНормалізованіПоказникиToolStripMenuItem_Click(object
sender, EventArgs e)
    {
        dataGridView2.Rows[0].Cells[0].Value = label1.Text;
        double w1 = double.Parse(textBox2.Text)*12 /
double.Parse(textBox1.Text);
        dataGridView2.Rows[0].Cells[1].Value = w1;
        dataGridView2.Rows[1].Cells[0].Value = label2.Text;
        double w2 = double.Parse(textBox4.Text) * 12 /
double.Parse(textBox3.Text);
        dataGridView2.Rows[1].Cells[1].Value = w2;
        dataGridView2.Rows[2].Cells[0].Value = label3.Text;
        double w3 = double.Parse(textBox6.Text) * 12 /
double.Parse(textBox5.Text);
        dataGridView2.Rows[2].Cells[1].Value = w3;
        dataGridView2.Rows[3].Cells[0].Value = label4.Text;
        double w4 = double.Parse(textBox8.Text) * 12 /
double.Parse(textBox7.Text);
        dataGridView2.Rows[3].Cells[1].Value = w4;
        dataGridView2.Rows[4].Cells[0].Value = label5.Text;
        double w5 = double.Parse(textBox10.Text) * 12 /
double.Parse(textBox9.Text);
        dataGridView2.Rows[4].Cells[1].Value = w5;
    }

```

```

        dataGridView2.Rows[5].Cells[0].Value = label6.Text;
        double w6 = double.Parse(textBox12.Text) * 12 /
double.Parse(textBox11.Text);
        dataGridView2.Rows[5].Cells[1].Value = w6;
        dataGridView2.Rows[6].Cells[0].Value = label7.Text;
        double w7 = double.Parse(textBox14.Text) * 12 /
double.Parse(textBox13.Text);
        dataGridView2.Rows[6].Cells[1].Value = w7;
        dataGridView2.Rows[7].Cells[0].Value = label8.Text;
        double w8 = double.Parse(textBox16.Text) * 12 /
double.Parse(textBox15.Text);
        dataGridView2.Rows[7].Cells[1].Value = w8;
        dataGridView2.Rows[8].Cells[0].Value = label9.Text;
        double w9 = double.Parse(textBox18.Text) * 12 /
double.Parse(textBox17.Text);
        dataGridView2.Rows[8].Cells[1].Value = w9;
        dataGridView2.Rows[9].Cells[0].Value = label10.Text;
        double w10 = double.Parse(textBox20.Text) * 12 /
double.Parse(textBox19.Text);
        dataGridView2.Rows[9].Cells[1].Value = w10;
        dataGridView2.Rows[10].Cells[0].Value =
label11.Text;
        double w11 = double.Parse(textBox22.Text) * 12 /
double.Parse(textBox21.Text);
        dataGridView2.Rows[10].Cells[1].Value = w11;
        dataGridView2.Rows[11].Cells[0].Value =
label12.Text;
        double w12 = double.Parse(textBox24.Text) * 12 /
double.Parse(textBox23.Text);
        dataGridView2.Rows[11].Cells[1].Value = w12;
        dataGridView2.Rows[12].Cells[0].Value =
label13.Text;
        double w13 = double.Parse(textBox26.Text) * 12 /
double.Parse(textBox25.Text);
        dataGridView2.Rows[12].Cells[1].Value = w13;
        dataGridView2.Rows[13].Cells[0].Value =
label14.Text;
        double w14 = double.Parse(textBox28.Text) * 12 /
double.Parse(textBox27.Text);
        dataGridView2.Rows[13].Cells[1].Value = w14;
        dataGridView2.Rows[14].Cells[0].Value =
label15.Text;
        double w15 = double.Parse(textBox30.Text) * 12 /
double.Parse(textBox29.Text);
        dataGridView2.Rows[14].Cells[1].Value = w15;
        dataGridView2.Rows[15].Cells[0].Value =
label16.Text;
        double w16 = double.Parse(textBox32.Text) * 12 /
double.Parse(textBox31.Text);
        dataGridView2.Rows[15].Cells[1].Value = w16;
        dataGridView2.Rows[16].Cells[0].Value =
label17.Text;

```

```

        double w17 = double.Parse(textBox34.Text) * 12 /
double.Parse(textBox33.Text);
        dataGridView2.Rows[16].Cells[1].Value = w17;
        dataGridView2.Rows[17].Cells[0].Value =
label18.Text;
        double w18 = double.Parse(textBox36.Text) * 12 /
double.Parse(textBox35.Text);
        dataGridView2.Rows[17].Cells[1].Value = w18;
    }
    private void
розрахуватиСреднєКвадратичнеВідхиленняToolStripMenuItem_Click(object sender, EventArgs e)
    {
        double w1 = double.Parse(textBox2.Text) * 12 /
double.Parse(textBox1.Text);
        double w2 = double.Parse(textBox4.Text) * 12 /
double.Parse(textBox3.Text);
        double w3 = double.Parse(textBox6.Text) * 12 /
double.Parse(textBox5.Text);
        double w4 = double.Parse(textBox8.Text) * 12 /
double.Parse(textBox7.Text);
        double w5 = double.Parse(textBox10.Text) * 12 /
double.Parse(textBox9.Text);
        double w6 = double.Parse(textBox12.Text) * 12 /
double.Parse(textBox11.Text);
        double w7 = double.Parse(textBox14.Text) * 12 /
double.Parse(textBox13.Text);
        double w8 = double.Parse(textBox16.Text) * 12 /
double.Parse(textBox15.Text);
        double w9 = double.Parse(textBox18.Text) * 12 /
double.Parse(textBox17.Text);
        double w10 = double.Parse(textBox20.Text) * 12 /
double.Parse(textBox19.Text);
        double w11 = double.Parse(textBox22.Text) * 12 /
double.Parse(textBox21.Text);
        double w12 = double.Parse(textBox24.Text) * 12 /
double.Parse(textBox23.Text);
        double w13 = double.Parse(textBox26.Text) * 12 /
double.Parse(textBox25.Text);
        double w14 = double.Parse(textBox28.Text) * 12 /
double.Parse(textBox27.Text);
        double w15 = double.Parse(textBox30.Text) * 12 /
double.Parse(textBox29.Text);
        double w16 = double.Parse(textBox32.Text) * 12 /
double.Parse(textBox31.Text);
        double w17 = double.Parse(textBox34.Text) * 12 /
double.Parse(textBox33.Text);
        double w18 = double.Parse(textBox36.Text) * 12 /
double.Parse(textBox35.Text);
        double g1 = 0.249, g2 = 0.064, g3 = 0.025, g4 =
0.238, g5 = 0.133, g6 = 0.150, g7 = 0.140;
        dataGridView3.Rows[0].Cells[1].Value = w1*g1;
        dataGridView3.Rows[1].Cells[1].Value = w2 * g1;
    }

```

```
dataGridView3.Rows[2].Cells[1].Value = w3 * g1;
dataGridView3.Rows[3].Cells[1].Value = w4 * g1;
dataGridView3.Rows[4].Cells[1].Value = w5 * g1;
dataGridView3.Rows[5].Cells[1].Value = w6 * g1;
dataGridView3.Rows[6].Cells[1].Value = w7 * g1;
dataGridView3.Rows[7].Cells[1].Value = w8 * g1;
dataGridView3.Rows[8].Cells[1].Value = w9 * g1;
dataGridView3.Rows[9].Cells[1].Value = w10 * g1;
dataGridView3.Rows[10].Cells[1].Value = w11 * g1;
dataGridView3.Rows[11].Cells[1].Value = w12 * g1;
dataGridView3.Rows[12].Cells[1].Value = w13 * g1;
dataGridView3.Rows[13].Cells[1].Value = w14 * g1;
dataGridView3.Rows[14].Cells[1].Value = w15 * g1;
dataGridView3.Rows[15].Cells[1].Value = w16 * g1;
dataGridView3.Rows[16].Cells[1].Value = w17 * g1;
dataGridView3.Rows[17].Cells[1].Value = w18 * g1;
dataGridView3.Rows[0].Cells[2].Value = w1 * g2;
dataGridView3.Rows[1].Cells[2].Value = w2 * g2;
dataGridView3.Rows[2].Cells[2].Value = w3 * g2;
dataGridView3.Rows[3].Cells[2].Value = w4 * g2;
dataGridView3.Rows[4].Cells[2].Value = w5 * g2;
dataGridView3.Rows[5].Cells[2].Value = w6 * g2;
dataGridView3.Rows[6].Cells[2].Value = w7 * g2;
dataGridView3.Rows[7].Cells[2].Value = w8 * g2;
dataGridView3.Rows[8].Cells[2].Value = w9 * g2;
dataGridView3.Rows[9].Cells[2].Value = w10 * g2;
dataGridView3.Rows[10].Cells[2].Value = w11 * g2;
dataGridView3.Rows[11].Cells[2].Value = w12 * g2;
dataGridView3.Rows[12].Cells[2].Value = w13 * g2;
dataGridView3.Rows[13].Cells[2].Value = w14 * g2;
dataGridView3.Rows[14].Cells[2].Value = w15 * g2;
dataGridView3.Rows[15].Cells[2].Value = w16 * g2;
dataGridView3.Rows[16].Cells[2].Value = w17 * g2;
dataGridView3.Rows[17].Cells[2].Value = w18 * g2;
dataGridView3.Rows[0].Cells[3].Value = w1 * g3;
dataGridView3.Rows[1].Cells[3].Value = w2 * g3;
dataGridView3.Rows[2].Cells[3].Value = w3 * g3;
dataGridView3.Rows[3].Cells[3].Value = w4 * g3;
dataGridView3.Rows[4].Cells[3].Value = w5 * g3;
dataGridView3.Rows[5].Cells[3].Value = w6 * g3;
dataGridView3.Rows[6].Cells[3].Value = w7 * g3;
dataGridView3.Rows[7].Cells[3].Value = w8 * g3;
dataGridView3.Rows[8].Cells[3].Value = w9 * g3;
dataGridView3.Rows[9].Cells[3].Value = w10 * g3;
dataGridView3.Rows[10].Cells[3].Value = w11 * g3;
dataGridView3.Rows[11].Cells[3].Value = w12 * g3;
dataGridView3.Rows[12].Cells[3].Value = w13 * g3;
dataGridView3.Rows[13].Cells[3].Value = w14 * g3;
dataGridView3.Rows[14].Cells[3].Value = w15 * g3;
dataGridView3.Rows[15].Cells[3].Value = w16 * g3;
dataGridView3.Rows[16].Cells[3].Value = w17 * g3;
dataGridView3.Rows[17].Cells[3].Value = w18 * g3;
dataGridView3.Rows[0].Cells[4].Value = w1 * g4;
```

```
dataGridView3.Rows[1].Cells[4].Value = w2 * g4;
dataGridView3.Rows[2].Cells[4].Value = w3 * g4;
dataGridView3.Rows[3].Cells[4].Value = w4 * g4;
dataGridView3.Rows[4].Cells[4].Value = w5 * g4;
dataGridView3.Rows[5].Cells[4].Value = w6 * g4;
dataGridView3.Rows[6].Cells[4].Value = w7 * g4;
dataGridView3.Rows[7].Cells[4].Value = w8 * g4;
dataGridView3.Rows[8].Cells[4].Value = w9 * g4;
dataGridView3.Rows[9].Cells[4].Value = w10 * g4;
dataGridView3.Rows[10].Cells[4].Value = w11 * g4;
dataGridView3.Rows[11].Cells[4].Value = w12 * g4;
dataGridView3.Rows[12].Cells[4].Value = w13 * g4;
dataGridView3.Rows[13].Cells[4].Value = w14 * g4;
dataGridView3.Rows[14].Cells[4].Value = w15 * g4;
dataGridView3.Rows[15].Cells[4].Value = w16 * g4;
dataGridView3.Rows[16].Cells[4].Value = w17 * g4;
dataGridView3.Rows[17].Cells[4].Value = w18 * g4;
dataGridView3.Rows[0].Cells[5].Value = w1 * g5;
dataGridView3.Rows[1].Cells[5].Value = w2 * g5;
dataGridView3.Rows[2].Cells[5].Value = w3 * g5;
dataGridView3.Rows[3].Cells[5].Value = w4 * g5;
dataGridView3.Rows[4].Cells[5].Value = w5 * g5;
dataGridView3.Rows[5].Cells[5].Value = w6 * g5;
dataGridView3.Rows[6].Cells[5].Value = w7 * g5;
dataGridView3.Rows[7].Cells[5].Value = w8 * g5;
dataGridView3.Rows[8].Cells[5].Value = w9 * g5;
dataGridView3.Rows[9].Cells[5].Value = w10 * g5;
dataGridView3.Rows[10].Cells[5].Value = w11 * g5;
dataGridView3.Rows[11].Cells[5].Value = w12 * g5;
dataGridView3.Rows[12].Cells[5].Value = w13 * g4;
dataGridView3.Rows[13].Cells[5].Value = w14 * g5;
dataGridView3.Rows[14].Cells[5].Value = w15 * g5;
dataGridView3.Rows[15].Cells[5].Value = w16 * g5;
dataGridView3.Rows[16].Cells[5].Value = w17 * g5;
dataGridView3.Rows[17].Cells[5].Value = w18 * g5;
dataGridView3.Rows[0].Cells[6].Value = w1 * g6;
dataGridView3.Rows[1].Cells[6].Value = w2 * g6;
dataGridView3.Rows[2].Cells[6].Value = w3 * g6;
dataGridView3.Rows[3].Cells[6].Value = w4 * g6;
dataGridView3.Rows[4].Cells[6].Value = w5 * g6;
dataGridView3.Rows[5].Cells[6].Value = w6 * g6;
dataGridView3.Rows[6].Cells[6].Value = w7 * g6;
dataGridView3.Rows[7].Cells[6].Value = w8 * g6;
dataGridView3.Rows[8].Cells[6].Value = w9 * g6;
dataGridView3.Rows[9].Cells[6].Value = w10 * g6;
dataGridView3.Rows[10].Cells[6].Value = w11 * g6;
dataGridView3.Rows[11].Cells[6].Value = w12 * g6;
dataGridView3.Rows[12].Cells[6].Value = w13 * g6;
dataGridView3.Rows[13].Cells[6].Value = w14 * g6;
dataGridView3.Rows[14].Cells[6].Value = w15 * g6;
dataGridView3.Rows[15].Cells[6].Value = w16 * g6;
dataGridView3.Rows[16].Cells[6].Value = w17 * g6;
dataGridView3.Rows[17].Cells[6].Value = w18 * g6;
```

```

dataGridView3.Rows[0].Cells[7].Value = w1 * g7;
dataGridView3.Rows[1].Cells[7].Value = w2 * g7;
dataGridView3.Rows[2].Cells[7].Value = w3 * g7;
dataGridView3.Rows[3].Cells[7].Value = w4 * g7;
dataGridView3.Rows[4].Cells[7].Value = w5 * g7;
dataGridView3.Rows[5].Cells[7].Value = w6 * g7;
dataGridView3.Rows[6].Cells[7].Value = w7 * g7;
dataGridView3.Rows[7].Cells[7].Value = w8 * g7;
dataGridView3.Rows[8].Cells[7].Value = w9 * g7;
dataGridView3.Rows[9].Cells[7].Value = w10 * g7;
dataGridView3.Rows[10].Cells[7].Value = w11 * g7;
dataGridView3.Rows[11].Cells[7].Value = w12 * g7;
dataGridView3.Rows[12].Cells[7].Value = w13 * g7;
dataGridView3.Rows[13].Cells[7].Value = w14 * g7;
dataGridView3.Rows[14].Cells[7].Value = w15 * g7;
dataGridView3.Rows[15].Cells[7].Value = w16 * g7;
dataGridView3.Rows[16].Cells[7].Value = w17 * g7;
dataGridView3.Rows[17].Cells[7].Value = w18 * g7;
    }
    private void
очиститиВсіПоляToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        for (int i = 1; i < dataGridView2.ColumnCount; i++)
        {
            for (int j = 0; j < dataGridView2.RowCount; j++)
            {
                dataGridView2.Rows[j].Cells[i].Value=" ";
            }
        }
        for (int i = 1; i < dataGridView3.ColumnCount; i++)
        {
            for (int j = 0; j < dataGridView3.RowCount; j++)
            {
                dataGridView2.Rows[j].Cells[i].Value = " ";
            }
        }
    }
    private void
зберегтиВExcelToolStripMenuItem_Click(object sender, EventArgs
e)
    {
        ExcelMicro::Application excelApplication = new
Microsoft.Office.Interop.Excel.Application();
        excelApplication.Visible = true;
        ExcelMicro::Workbook excelBook =
excelApplication.Workbooks.Add(1);
        ExcelMicro::Sheets excelSheet =
excelBook.Worksheets;
        ExcelMicro::Worksheet excelWorksheet =
(ExcelMicro::Worksheet)excelSheet.get_Item("Лист1");
        ExcelMicro::Range excelCell =
excelWorksheet.get_Range(excelWorksheet.Cells[1, 1],

```



```

excelWorksheet.Cells[dataGridView2.RowCount + 1,
dataGridView2.ColumnCount];
    excelCell.NumberFormat = "@";
    object[,] dataGrdVw = new
object[dataGridView2.RowCount+1 , dataGridView2.ColumnCount];
    for (int i = 0; i < dataGridView2.RowCount; i++)
    {
        for (int j = 0; j < dataGridView2.ColumnCount;
j++)
            {
                dataGrdVw[i, j] =
dataGridView2.Rows[i].Cells[j].Value;
                //dataGrdVw[0, 0] = 1;
            }
    }
excelCell.set_Value(ExcelMicro::XlRangeValueType.xlRangeValu
eDefault, dataGrdVw);
    excelCell.EntireColumn.AutoFit();
    excelCell.Font.ColorIndex = 1;
    Microsoft.Office.Interop.Excel.XlBorderWeight
xlBorder = new Microsoft.Office.Interop.Excel.XlBorderWeight();
    Microsoft.Office.Interop.Excel.XlColorIndex xlColor
=
Microsoft.Office.Interop.Excel.XlColorIndex.xlColorIndexAutomati
c;
    xlBorder =
Microsoft.Office.Interop.Excel.XlBorderWeight.xlThick;
    excelCell.BorderAround(7, xlBorder, xlColor, 1);
}

private void
заНормованимиФінансовимиПоказникамиToolStripMenuItem_Click(object sender, EventArgs e)
{
    double w1 = double.Parse(textBox2.Text) * 12 /
double.Parse(textBox1.Text);
    double w2 = double.Parse(textBox4.Text) * 12 /
double.Parse(textBox3.Text);
    double w3 = double.Parse(textBox6.Text) * 12 /
double.Parse(textBox5.Text);
    double w4 = double.Parse(textBox8.Text) * 12 /
double.Parse(textBox7.Text);
    double w5 = double.Parse(textBox10.Text) * 12 /
double.Parse(textBox9.Text);
    double w6 = double.Parse(textBox12.Text) * 12 /
double.Parse(textBox11.Text);
    double w7 = double.Parse(textBox14.Text) * 12 /
double.Parse(textBox13.Text);
    double w8 = double.Parse(textBox16.Text) * 12 /
double.Parse(textBox15.Text);
    double w9 = double.Parse(textBox18.Text) * 12 /
double.Parse(textBox17.Text);
    double w10 = double.Parse(textBox20.Text) * 12 /
double.Parse(textBox19.Text);
}

```

```

        double w11 = double.Parse(textBox22.Text) * 12 /
double.Parse(textBox21.Text);
        double w12 = double.Parse(textBox24.Text) * 12 /
double.Parse(textBox23.Text);
        double w13 = double.Parse(textBox26.Text) * 12 /
double.Parse(textBox25.Text);
        double w14 = double.Parse(textBox28.Text) * 12 /
double.Parse(textBox27.Text);
        double w15 = double.Parse(textBox30.Text) * 12 /
double.Parse(textBox29.Text);
        double w16 = double.Parse(textBox32.Text) * 12 /
double.Parse(textBox31.Text);
        double w17 = double.Parse(textBox34.Text) * 12 /
double.Parse(textBox33.Text);
        double w18 = double.Parse(textBox36.Text) * 12 /
double.Parse(textBox35.Text);
        double g1 = 0.249, g2 = 0.064, g3 = 0.025, g4 =
0.238, g5 = 0.133, g6 = 0.150, g7 = 0.140;

```

```

dataGridView3.Rows[0].Cells[1].Value = w1 * g1;
dataGridView3.Rows[1].Cells[1].Value = w2 * g1;
dataGridView3.Rows[2].Cells[1].Value = w3 * g1;
dataGridView3.Rows[3].Cells[1].Value = w4 * g1;
dataGridView3.Rows[4].Cells[1].Value = w5 * g1;
dataGridView3.Rows[5].Cells[1].Value = w6 * g1;
dataGridView3.Rows[6].Cells[1].Value = w7 * g1;
dataGridView3.Rows[7].Cells[1].Value = w8 * g1;
dataGridView3.Rows[8].Cells[1].Value = w9 * g1;
dataGridView3.Rows[9].Cells[1].Value = w10 * g1;
dataGridView3.Rows[10].Cells[1].Value = w11 * g1;
dataGridView3.Rows[11].Cells[1].Value = w12 * g1;
dataGridView3.Rows[12].Cells[1].Value = w13 * g1;
dataGridView3.Rows[13].Cells[1].Value = w14 * g1;
dataGridView3.Rows[14].Cells[1].Value = w15 * g1;
dataGridView3.Rows[15].Cells[1].Value = w16 * g1;
dataGridView3.Rows[16].Cells[1].Value = w17 * g1;
dataGridView3.Rows[17].Cells[1].Value = w18 * g1;
dataGridView3.Rows[0].Cells[2].Value = w1 * g2;
dataGridView3.Rows[1].Cells[2].Value = w2 * g2;
dataGridView3.Rows[2].Cells[2].Value = w3 * g2;
dataGridView3.Rows[3].Cells[2].Value = w4 * g2;
dataGridView3.Rows[4].Cells[2].Value = w5 * g2;
dataGridView3.Rows[5].Cells[2].Value = w6 * g2;
dataGridView3.Rows[6].Cells[2].Value = w7 * g2;
dataGridView3.Rows[7].Cells[2].Value = w8 * g2;
dataGridView3.Rows[8].Cells[2].Value = w9 * g2;
dataGridView3.Rows[9].Cells[2].Value = w10 * g2;
dataGridView3.Rows[10].Cells[2].Value = w11 * g2;
dataGridView3.Rows[11].Cells[2].Value = w12 * g2;
dataGridView3.Rows[12].Cells[2].Value = w13 * g2;
dataGridView3.Rows[13].Cells[2].Value = w14 * g2;
dataGridView3.Rows[14].Cells[2].Value = w15 * g2;
dataGridView3.Rows[15].Cells[2].Value = w16 * g2;

```

```
dataGridView3.Rows[16].Cells[2].Value = w17 * g2;
dataGridView3.Rows[17].Cells[2].Value = w18 * g2;
dataGridView3.Rows[0].Cells[3].Value = w1 * g3;
dataGridView3.Rows[1].Cells[3].Value = w2 * g3;
dataGridView3.Rows[2].Cells[3].Value = w3 * g3;
dataGridView3.Rows[3].Cells[3].Value = w4 * g3;
dataGridView3.Rows[4].Cells[3].Value = w5 * g3;
dataGridView3.Rows[5].Cells[3].Value = w6 * g3;
dataGridView3.Rows[6].Cells[3].Value = w7 * g3;
dataGridView3.Rows[7].Cells[3].Value = w8 * g3;
dataGridView3.Rows[8].Cells[3].Value = w9 * g3;
dataGridView3.Rows[9].Cells[3].Value = w10 * g3;
dataGridView3.Rows[10].Cells[3].Value = w11 * g3;
dataGridView3.Rows[11].Cells[3].Value = w12 * g3;
dataGridView3.Rows[12].Cells[3].Value = w13 * g3;
dataGridView3.Rows[13].Cells[3].Value = w14 * g3;
dataGridView3.Rows[14].Cells[3].Value = w15 * g3;
dataGridView3.Rows[15].Cells[3].Value = w16 * g3;
dataGridView3.Rows[16].Cells[3].Value = w17 * g3;
dataGridView3.Rows[17].Cells[3].Value = w18 * g3;
dataGridView3.Rows[0].Cells[4].Value = w1 * g4;
dataGridView3.Rows[1].Cells[4].Value = w2 * g4;
dataGridView3.Rows[2].Cells[4].Value = w3 * g4;
dataGridView3.Rows[3].Cells[4].Value = w4 * g4;
dataGridView3.Rows[4].Cells[4].Value = w5 * g4;
dataGridView3.Rows[5].Cells[4].Value = w6 * g4;
dataGridView3.Rows[6].Cells[4].Value = w7 * g4;
dataGridView3.Rows[7].Cells[4].Value = w8 * g4;
dataGridView3.Rows[8].Cells[4].Value = w9 * g4;
dataGridView3.Rows[9].Cells[4].Value = w10 * g4;
dataGridView3.Rows[10].Cells[4].Value = w11 * g4;
dataGridView3.Rows[11].Cells[4].Value = w12 * g4;
dataGridView3.Rows[12].Cells[4].Value = w13 * g4;
dataGridView3.Rows[13].Cells[4].Value = w14 * g4;
dataGridView3.Rows[14].Cells[4].Value = w15 * g4;
dataGridView3.Rows[15].Cells[4].Value = w16 * g4;
dataGridView3.Rows[16].Cells[4].Value = w17 * g4;
dataGridView3.Rows[17].Cells[4].Value = w18 * g4;
dataGridView3.Rows[0].Cells[5].Value = w1 * g5;
dataGridView3.Rows[1].Cells[5].Value = w2 * g5;
dataGridView3.Rows[2].Cells[5].Value = w3 * g5;
dataGridView3.Rows[3].Cells[5].Value = w4 * g5;
dataGridView3.Rows[4].Cells[5].Value = w5 * g5;
dataGridView3.Rows[5].Cells[5].Value = w6 * g5;
dataGridView3.Rows[6].Cells[5].Value = w7 * g5;
dataGridView3.Rows[7].Cells[5].Value = w8 * g5;
dataGridView3.Rows[8].Cells[5].Value = w9 * g5;
dataGridView3.Rows[9].Cells[5].Value = w10 * g5;
dataGridView3.Rows[10].Cells[5].Value = w11 * g5;
dataGridView3.Rows[11].Cells[5].Value = w12 * g5;
dataGridView3.Rows[12].Cells[5].Value = w13 * g4;
dataGridView3.Rows[13].Cells[5].Value = w14 * g5;
dataGridView3.Rows[14].Cells[5].Value = w15 * g5;
```

```

dataGridView3.Rows[15].Cells[5].Value = w16 * g5;
dataGridView3.Rows[16].Cells[5].Value = w17 * g5;
dataGridView3.Rows[17].Cells[5].Value = w18 * g5;
dataGridView3.Rows[0].Cells[6].Value = w1 * g6;
dataGridView3.Rows[1].Cells[6].Value = w2 * g6;
dataGridView3.Rows[2].Cells[6].Value = w3 * g6;
dataGridView3.Rows[3].Cells[6].Value = w4 * g6;
dataGridView3.Rows[4].Cells[6].Value = w5 * g6;
dataGridView3.Rows[5].Cells[6].Value = w6 * g6;
dataGridView3.Rows[6].Cells[6].Value = w7 * g6;
dataGridView3.Rows[7].Cells[6].Value = w8 * g6;
dataGridView3.Rows[8].Cells[6].Value = w9 * g6;
dataGridView3.Rows[9].Cells[6].Value = w10 * g6;
dataGridView3.Rows[10].Cells[6].Value = w11 * g6;
dataGridView3.Rows[11].Cells[6].Value = w12 * g6;
dataGridView3.Rows[12].Cells[6].Value = w13 * g6;
dataGridView3.Rows[13].Cells[6].Value = w14 * g6;
dataGridView3.Rows[14].Cells[6].Value = w15 * g6;
dataGridView3.Rows[15].Cells[6].Value = w16 * g6;
dataGridView3.Rows[16].Cells[6].Value = w17 * g6;
dataGridView3.Rows[17].Cells[6].Value = w18 * g6;
dataGridView3.Rows[0].Cells[7].Value = w1 * g7;
dataGridView3.Rows[1].Cells[7].Value = w2 * g7;
dataGridView3.Rows[2].Cells[7].Value = w3 * g7;
dataGridView3.Rows[3].Cells[7].Value = w4 * g7;
dataGridView3.Rows[4].Cells[7].Value = w5 * g7;
dataGridView3.Rows[5].Cells[7].Value = w6 * g7;
dataGridView3.Rows[6].Cells[7].Value = w7 * g7;
dataGridView3.Rows[7].Cells[7].Value = w8 * g7;
dataGridView3.Rows[8].Cells[7].Value = w9 * g7;
dataGridView3.Rows[9].Cells[7].Value = w10 * g7;
dataGridView3.Rows[10].Cells[7].Value = w11 * g7;
dataGridView3.Rows[11].Cells[7].Value = w12 * g7;
dataGridView3.Rows[12].Cells[7].Value = w13 * g7;
dataGridView3.Rows[13].Cells[7].Value = w14 * g7;
dataGridView3.Rows[14].Cells[7].Value = w15 * g7;
dataGridView3.Rows[15].Cells[7].Value = w16 * g7;
dataGridView3.Rows[16].Cells[7].Value = w17 * g7;
dataGridView3.Rows[17].Cells[7].Value = w18 * g7;
}
private void
заМісячнимиФінансовимиПоказникамиToolStripMenuItem_Click(object
sender, EventArgs e)
{
    double w1 = double.Parse(textBox2.Text);
    double w2 = double.Parse(textBox4.Text);
    double w3 = double.Parse(textBox6.Text);
    double w4 = double.Parse(textBox8.Text);
    double w5 = double.Parse(textBox10.Text);
    double w6 = double.Parse(textBox12.Text);
    double w7 = double.Parse(textBox14.Text);
    double w8 = double.Parse(textBox16.Text);
}

```

```

double w9 = double.Parse(textBox18.Text) ;
double w10 = double.Parse(textBox20.Text) ;
double w11 = double.Parse(textBox22.Text) ;
double w12 = double.Parse(textBox24.Text) ;
double w13 = double.Parse(textBox26.Text) ;
double w14 = double.Parse(textBox28.Text) ;
double w15 = double.Parse(textBox30.Text) ;
double w16 = double.Parse(textBox32.Text) ;
double w17 = double.Parse(textBox34.Text) ;
double w18 = double.Parse(textBox36.Text) ;
double g1 = 0.249, g2 = 0.064, g3 = 0.025, g4 =
0.238, g5 = 0.133, g6 = 0.150, g7 = 0.140;
dataGridView3.Rows[0].Cells[1].Value = w1 * g1;
dataGridView3.Rows[1].Cells[1].Value = w2 * g1;
dataGridView3.Rows[2].Cells[1].Value = w3 * g1;
dataGridView3.Rows[3].Cells[1].Value = w4 * g1;
dataGridView3.Rows[4].Cells[1].Value = w5 * g1;
dataGridView3.Rows[5].Cells[1].Value = w6 * g1;
dataGridView3.Rows[6].Cells[1].Value = w7 * g1;
dataGridView3.Rows[7].Cells[1].Value = w8 * g1;
dataGridView3.Rows[8].Cells[1].Value = w9 * g1;
dataGridView3.Rows[9].Cells[1].Value = w10 * g1;
dataGridView3.Rows[10].Cells[1].Value = w11 * g1;
dataGridView3.Rows[11].Cells[1].Value = w12 * g1;
dataGridView3.Rows[12].Cells[1].Value = w13 * g1;
dataGridView3.Rows[13].Cells[1].Value = w14 * g1;
dataGridView3.Rows[14].Cells[1].Value = w15 * g1;
dataGridView3.Rows[15].Cells[1].Value = w16 * g1;
dataGridView3.Rows[16].Cells[1].Value = w17 * g1;
dataGridView3.Rows[17].Cells[1].Value = w18 * g1;
dataGridView3.Rows[0].Cells[2].Value = w1 * g2;
dataGridView3.Rows[1].Cells[2].Value = w2 * g2;
dataGridView3.Rows[2].Cells[2].Value = w3 * g2;
dataGridView3.Rows[3].Cells[2].Value = w4 * g2;
dataGridView3.Rows[4].Cells[2].Value = w5 * g2;
dataGridView3.Rows[5].Cells[2].Value = w6 * g2;
dataGridView3.Rows[6].Cells[2].Value = w7 * g2;
dataGridView3.Rows[7].Cells[2].Value = w8 * g2;
dataGridView3.Rows[8].Cells[2].Value = w9 * g2;
dataGridView3.Rows[9].Cells[2].Value = w10 * g2;
dataGridView3.Rows[10].Cells[2].Value = w11 * g2;
dataGridView3.Rows[11].Cells[2].Value = w12 * g2;
dataGridView3.Rows[12].Cells[2].Value = w13 * g2;
dataGridView3.Rows[13].Cells[2].Value = w14 * g2;
dataGridView3.Rows[14].Cells[2].Value = w15 * g2;
dataGridView3.Rows[15].Cells[2].Value = w16 * g2;
dataGridView3.Rows[16].Cells[2].Value = w17 * g2;
dataGridView3.Rows[17].Cells[2].Value = w18 * g2;
dataGridView3.Rows[0].Cells[3].Value = w1 * g3;
dataGridView3.Rows[1].Cells[3].Value = w2 * g3;
dataGridView3.Rows[2].Cells[3].Value = w3 * g3;
dataGridView3.Rows[3].Cells[3].Value = w4 * g3;
dataGridView3.Rows[4].Cells[3].Value = w5 * g3;

```

```
dataGridView3.Rows[5].Cells[3].Value = w6 * g3;
dataGridView3.Rows[6].Cells[3].Value = w7 * g3;
dataGridView3.Rows[7].Cells[3].Value = w8 * g3;
dataGridView3.Rows[8].Cells[3].Value = w9 * g3;
dataGridView3.Rows[9].Cells[3].Value = w10 * g3;
dataGridView3.Rows[10].Cells[3].Value = w11 * g3;
dataGridView3.Rows[11].Cells[3].Value = w12 * g3;
dataGridView3.Rows[12].Cells[3].Value = w13 * g3;
dataGridView3.Rows[13].Cells[3].Value = w14 * g3;
dataGridView3.Rows[14].Cells[3].Value = w15 * g3;
dataGridView3.Rows[15].Cells[3].Value = w16 * g3;
dataGridView3.Rows[16].Cells[3].Value = w17 * g3;
dataGridView3.Rows[17].Cells[3].Value = w18 * g3;
dataGridView3.Rows[0].Cells[4].Value = w1 * g4;
dataGridView3.Rows[1].Cells[4].Value = w2 * g4;
dataGridView3.Rows[2].Cells[4].Value = w3 * g4;
dataGridView3.Rows[3].Cells[4].Value = w4 * g4;
dataGridView3.Rows[4].Cells[4].Value = w5 * g4;
dataGridView3.Rows[5].Cells[4].Value = w6 * g4;
dataGridView3.Rows[6].Cells[4].Value = w7 * g4;
dataGridView3.Rows[7].Cells[4].Value = w8 * g4;
dataGridView3.Rows[8].Cells[4].Value = w9 * g4;
dataGridView3.Rows[9].Cells[4].Value = w10 * g4;
dataGridView3.Rows[10].Cells[4].Value = w11 * g4;
dataGridView3.Rows[11].Cells[4].Value = w12 * g4;
dataGridView3.Rows[12].Cells[4].Value = w13 * g4;
dataGridView3.Rows[13].Cells[4].Value = w14 * g4;
dataGridView3.Rows[14].Cells[4].Value = w15 * g4;
dataGridView3.Rows[15].Cells[4].Value = w16 * g4;
dataGridView3.Rows[16].Cells[4].Value = w17 * g4;
dataGridView3.Rows[17].Cells[4].Value = w18 * g4;
dataGridView3.Rows[0].Cells[5].Value = w1 * g5;
dataGridView3.Rows[1].Cells[5].Value = w2 * g5;
dataGridView3.Rows[2].Cells[5].Value = w3 * g5;
dataGridView3.Rows[3].Cells[5].Value = w4 * g5;
dataGridView3.Rows[4].Cells[5].Value = w5 * g5;
dataGridView3.Rows[5].Cells[5].Value = w6 * g5;
dataGridView3.Rows[6].Cells[5].Value = w7 * g5;
dataGridView3.Rows[7].Cells[5].Value = w8 * g5;
dataGridView3.Rows[8].Cells[5].Value = w9 * g5;
dataGridView3.Rows[9].Cells[5].Value = w10 * g5;
dataGridView3.Rows[10].Cells[5].Value = w11 * g5;
dataGridView3.Rows[11].Cells[5].Value = w12 * g5;
dataGridView3.Rows[12].Cells[5].Value = w13 * g4;
dataGridView3.Rows[13].Cells[5].Value = w14 * g5;
dataGridView3.Rows[14].Cells[5].Value = w15 * g5;
dataGridView3.Rows[15].Cells[5].Value = w16 * g5;
dataGridView3.Rows[16].Cells[5].Value = w17 * g5;
dataGridView3.Rows[17].Cells[5].Value = w18 * g5;
dataGridView3.Rows[0].Cells[6].Value = w1 * g6;
dataGridView3.Rows[1].Cells[6].Value = w2 * g6;
dataGridView3.Rows[2].Cells[6].Value = w3 * g6;
dataGridView3.Rows[3].Cells[6].Value = w4 * g6;
```

```

dataGridView3.Rows[4].Cells[6].Value = w5 * g6;
dataGridView3.Rows[5].Cells[6].Value = w6 * g6;
dataGridView3.Rows[6].Cells[6].Value = w7 * g6;
dataGridView3.Rows[7].Cells[6].Value = w8 * g6;
dataGridView3.Rows[8].Cells[6].Value = w9 * g6;
dataGridView3.Rows[9].Cells[6].Value = w10 * g6;
dataGridView3.Rows[10].Cells[6].Value = w11 * g6;
dataGridView3.Rows[11].Cells[6].Value = w12 * g6;
dataGridView3.Rows[12].Cells[6].Value = w13 * g6;
dataGridView3.Rows[13].Cells[6].Value = w14 * g6;
dataGridView3.Rows[14].Cells[6].Value = w15 * g6;
dataGridView3.Rows[15].Cells[6].Value = w16 * g6;
dataGridView3.Rows[16].Cells[6].Value = w17 * g6;
dataGridView3.Rows[17].Cells[6].Value = w18 * g6;
dataGridView3.Rows[0].Cells[7].Value = w1 * g7;
dataGridView3.Rows[1].Cells[7].Value = w2 * g7;
dataGridView3.Rows[2].Cells[7].Value = w3 * g7;
dataGridView3.Rows[3].Cells[7].Value = w4 * g7;
dataGridView3.Rows[4].Cells[7].Value = w5 * g7;
dataGridView3.Rows[5].Cells[7].Value = w6 * g7;
dataGridView3.Rows[6].Cells[7].Value = w7 * g7;
dataGridView3.Rows[7].Cells[7].Value = w8 * g7;
dataGridView3.Rows[8].Cells[7].Value = w9 * g7;
dataGridView3.Rows[9].Cells[7].Value = w10 * g7;
dataGridView3.Rows[10].Cells[7].Value = w11 * g7;
dataGridView3.Rows[11].Cells[7].Value = w12 * g7;
dataGridView3.Rows[12].Cells[7].Value = w13 * g7;
dataGridView3.Rows[13].Cells[7].Value = w14 * g7;
dataGridView3.Rows[14].Cells[7].Value = w15 * g7;
dataGridView3.Rows[15].Cells[7].Value = w16 * g7;
dataGridView3.Rows[16].Cells[7].Value = w17 * g7;
dataGridView3.Rows[17].Cells[7].Value = w18 * g7;
}
private void
заРічнимиФінансовимиПоказникамиToolStripMenuItem_Click(object
sender, EventArgs e)
{
    double w1 = double.Parse(textBox1.Text);
    double w2 = double.Parse(textBox3.Text);
    double w3 = double.Parse(textBox5.Text);
    double w4 = double.Parse(textBox7.Text);
    double w5 = double.Parse(textBox9.Text);
    double w6 = double.Parse(textBox11.Text);
    double w7 = double.Parse(textBox13.Text);
    double w8 = double.Parse(textBox15.Text);
    double w9 = double.Parse(textBox17.Text);
    double w10 = double.Parse(textBox19.Text);
    double w11 = double.Parse(textBox21.Text);
    double w12 = double.Parse(textBox23.Text);
    double w13 = double.Parse(textBox25.Text);
    double w14 = double.Parse(textBox27.Text);
    double w15 = double.Parse(textBox29.Text);
    double w16 = double.Parse(textBox31.Text);

```



```

double w17 = double.Parse(textBox33.Text);
double w18 = double.Parse(textBox35.Text);
double g1 = 0.249, g2 = 0.064, g3 = 0.025, g4 =
0.238, g5 = 0.133, g6 = 0.150, g7 = 0.140;
dataGridView3.Rows[0].Cells[1].Value = w1 * g1;
dataGridView3.Rows[1].Cells[1].Value = w2 * g1;
dataGridView3.Rows[2].Cells[1].Value = w3 * g1;
dataGridView3.Rows[3].Cells[1].Value = w4 * g1;
dataGridView3.Rows[4].Cells[1].Value = w5 * g1;
dataGridView3.Rows[5].Cells[1].Value = w6 * g1;
dataGridView3.Rows[6].Cells[1].Value = w7 * g1;
dataGridView3.Rows[7].Cells[1].Value = w8 * g1;
dataGridView3.Rows[8].Cells[1].Value = w9 * g1;
dataGridView3.Rows[9].Cells[1].Value = w10 * g1;
dataGridView3.Rows[10].Cells[1].Value = w11 * g1;
dataGridView3.Rows[11].Cells[1].Value = w12 * g1;
dataGridView3.Rows[12].Cells[1].Value = w13 * g1;
dataGridView3.Rows[13].Cells[1].Value = w14 * g1;
dataGridView3.Rows[14].Cells[1].Value = w15 * g1;
dataGridView3.Rows[15].Cells[1].Value = w16 * g1;
dataGridView3.Rows[16].Cells[1].Value = w17 * g1;
dataGridView3.Rows[17].Cells[1].Value = w18 * g1;
dataGridView3.Rows[0].Cells[2].Value = w1 * g2;
dataGridView3.Rows[1].Cells[2].Value = w2 * g2;
dataGridView3.Rows[2].Cells[2].Value = w3 * g2;
dataGridView3.Rows[3].Cells[2].Value = w4 * g2;
dataGridView3.Rows[4].Cells[2].Value = w5 * g2;
dataGridView3.Rows[5].Cells[2].Value = w6 * g2;
dataGridView3.Rows[6].Cells[2].Value = w7 * g2;
dataGridView3.Rows[7].Cells[2].Value = w8 * g2;
dataGridView3.Rows[8].Cells[2].Value = w9 * g2;
dataGridView3.Rows[9].Cells[2].Value = w10 * g2;
dataGridView3.Rows[10].Cells[2].Value = w11 * g2;
dataGridView3.Rows[11].Cells[2].Value = w12 * g2;
dataGridView3.Rows[12].Cells[2].Value = w13 * g2;
dataGridView3.Rows[13].Cells[2].Value = w14 * g2;
dataGridView3.Rows[14].Cells[2].Value = w15 * g2;
dataGridView3.Rows[15].Cells[2].Value = w16 * g2;
dataGridView3.Rows[16].Cells[2].Value = w17 * g2;
dataGridView3.Rows[17].Cells[2].Value = w18 * g2;
dataGridView3.Rows[0].Cells[3].Value = w1 * g3;
dataGridView3.Rows[1].Cells[3].Value = w2 * g3;
dataGridView3.Rows[2].Cells[3].Value = w3 * g3;
dataGridView3.Rows[3].Cells[3].Value = w4 * g3;
dataGridView3.Rows[4].Cells[3].Value = w5 * g3;
dataGridView3.Rows[5].Cells[3].Value = w6 * g3;
dataGridView3.Rows[6].Cells[3].Value = w7 * g3;
dataGridView3.Rows[7].Cells[3].Value = w8 * g3;
dataGridView3.Rows[8].Cells[3].Value = w9 * g3;
dataGridView3.Rows[9].Cells[3].Value = w10 * g3;
dataGridView3.Rows[10].Cells[3].Value = w11 * g3;
dataGridView3.Rows[11].Cells[3].Value = w12 * g3;
dataGridView3.Rows[12].Cells[3].Value = w13 * g3;

```



```
dataGridView3.Rows[13].Cells[3].Value = w14 * g3;
dataGridView3.Rows[14].Cells[3].Value = w15 * g3;
dataGridView3.Rows[15].Cells[3].Value = w16 * g3;
dataGridView3.Rows[16].Cells[3].Value = w17 * g3;
dataGridView3.Rows[17].Cells[3].Value = w18 * g3;
dataGridView3.Rows[0].Cells[4].Value = w1 * g4;
dataGridView3.Rows[1].Cells[4].Value = w2 * g4;
dataGridView3.Rows[2].Cells[4].Value = w3 * g4;
dataGridView3.Rows[3].Cells[4].Value = w4 * g4;
dataGridView3.Rows[4].Cells[4].Value = w5 * g4;
dataGridView3.Rows[5].Cells[4].Value = w6 * g4;
dataGridView3.Rows[6].Cells[4].Value = w7 * g4;
dataGridView3.Rows[7].Cells[4].Value = w8 * g4;
dataGridView3.Rows[8].Cells[4].Value = w9 * g4;
dataGridView3.Rows[9].Cells[4].Value = w10 * g4;
dataGridView3.Rows[10].Cells[4].Value = w11 * g4;
dataGridView3.Rows[11].Cells[4].Value = w12 * g4;
dataGridView3.Rows[12].Cells[4].Value = w13 * g4;
dataGridView3.Rows[13].Cells[4].Value = w14 * g4;
dataGridView3.Rows[14].Cells[4].Value = w15 * g4;
dataGridView3.Rows[15].Cells[4].Value = w16 * g4;
dataGridView3.Rows[16].Cells[4].Value = w17 * g4;
dataGridView3.Rows[17].Cells[4].Value = w18 * g4;
dataGridView3.Rows[0].Cells[5].Value = w1 * g5;
dataGridView3.Rows[1].Cells[5].Value = w2 * g5;
dataGridView3.Rows[2].Cells[5].Value = w3 * g5;
dataGridView3.Rows[3].Cells[5].Value = w4 * g5;
dataGridView3.Rows[4].Cells[5].Value = w5 * g5;
dataGridView3.Rows[5].Cells[5].Value = w6 * g5;
dataGridView3.Rows[6].Cells[5].Value = w7 * g5;
dataGridView3.Rows[7].Cells[5].Value = w8 * g5;
dataGridView3.Rows[8].Cells[5].Value = w9 * g5;
dataGridView3.Rows[9].Cells[5].Value = w10 * g5;
dataGridView3.Rows[10].Cells[5].Value = w11 * g5;
dataGridView3.Rows[11].Cells[5].Value = w12 * g5;
dataGridView3.Rows[12].Cells[5].Value = w13 * g4;
dataGridView3.Rows[13].Cells[5].Value = w14 * g5;
dataGridView3.Rows[14].Cells[5].Value = w15 * g5;
dataGridView3.Rows[15].Cells[5].Value = w16 * g5;
dataGridView3.Rows[16].Cells[5].Value = w17 * g5;
dataGridView3.Rows[17].Cells[5].Value = w18 * g5;
dataGridView3.Rows[0].Cells[6].Value = w1 * g6;
dataGridView3.Rows[1].Cells[6].Value = w2 * g6;
dataGridView3.Rows[2].Cells[6].Value = w3 * g6;
dataGridView3.Rows[3].Cells[6].Value = w4 * g6;
dataGridView3.Rows[4].Cells[6].Value = w5 * g6;
dataGridView3.Rows[5].Cells[6].Value = w6 * g6;
dataGridView3.Rows[6].Cells[6].Value = w7 * g6;
dataGridView3.Rows[7].Cells[6].Value = w8 * g6;
dataGridView3.Rows[8].Cells[6].Value = w9 * g6;
dataGridView3.Rows[9].Cells[6].Value = w10 * g6;
dataGridView3.Rows[10].Cells[6].Value = w11 * g6;
dataGridView3.Rows[11].Cells[6].Value = w12 * g6;
```

```
dataGridView3.Rows[12].Cells[6].Value = w13 * g6;  
dataGridView3.Rows[13].Cells[6].Value = w14 * g6;  
dataGridView3.Rows[14].Cells[6].Value = w15 * g6;  
dataGridView3.Rows[15].Cells[6].Value = w16 * g6;  
dataGridView3.Rows[16].Cells[6].Value = w17 * g6;  
dataGridView3.Rows[17].Cells[6].Value = w18 * g6;  
dataGridView3.Rows[0].Cells[7].Value = w1 * g7;  
dataGridView3.Rows[1].Cells[7].Value = w2 * g7;  
dataGridView3.Rows[2].Cells[7].Value = w3 * g7;  
dataGridView3.Rows[3].Cells[7].Value = w4 * g7;  
dataGridView3.Rows[4].Cells[7].Value = w5 * g7;  
dataGridView3.Rows[5].Cells[7].Value = w6 * g7;  
dataGridView3.Rows[6].Cells[7].Value = w7 * g7;  
dataGridView3.Rows[7].Cells[7].Value = w8 * g7;  
dataGridView3.Rows[8].Cells[7].Value = w9 * g7;  
dataGridView3.Rows[9].Cells[7].Value = w10 * g7;  
dataGridView3.Rows[10].Cells[7].Value = w11 * g7;  
dataGridView3.Rows[11].Cells[7].Value = w12 * g7;  
dataGridView3.Rows[12].Cells[7].Value = w13 * g7;  
dataGridView3.Rows[13].Cells[7].Value = w14 * g7;  
dataGridView3.Rows[14].Cells[7].Value = w15 * g7;  
dataGridView3.Rows[15].Cells[7].Value = w16 * g7;  
dataGridView3.Rows[16].Cells[7].Value = w17 * g7;  
dataGridView3.Rows[17].Cells[7].Value = w18 * g7;  
    }  
}  
}
```

## Лістинг В.2 – Код другої форми

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace WindowsFormsApplication1
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }
    }
}
```

## Лістинг В.3 – Код по аналізу операційного ризику

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.IO;
using ZedGraph;

namespace Lab5
{
    public partial class Labs : Form
    {
        public Labs()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            string path = "";
            OpenFileDialog ofd = new OpenFileDialog();
            ofd.InitialDirectory = "c:\\";
            ofd.Filter = "txt files (*.txt)|*.txt|All files
(*.*)|*.*";
            ofd.FilterIndex = 1;
            ofd.RestoreDirectory = true;
            if (ofd.ShowDialog() == DialogResult.OK)
            {
                path = ofd.FileName;
                string[] lines = File.ReadAllLines(path);
            }
            dataGridView1.Rows.Clear();
            string inpstr;
            if (File.Exists(path))
            {
                StreamReader rd = new StreamReader(path);
                string[] str;
                char[] delim = new char[] { ' ' }; //Разделители
                while ((inpstr = rd.ReadLine()) != null)
                {
                    str = inpstr.Split(delim); //Разбиваем строку
на отдельные строки по разделителям
                    dataGridView1.Rows.Add(str);
                }
                rd.Close();
            }
        }
    }
}

```

```

    }
}

private void button4_Click(object sender, EventArgs e)
{
    if (c1.Text == "сукупний")
    {
        GraphPane myPane = zgl.GraphPane;
        string l = "вирогідність";
        myPane.Title.Text = l;
        myPane.XAxis.Title.Text = "роки";
        myPane.YAxis.Title.Text = "вирогідності";
        double x, y;
        double[] a = new double[dataGridView1.RowCount];
        double[] b = new double[dataGridView1.RowCount];
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[1].Value);
            b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);
        }
        PointPairList list1 = new PointPairList();
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            x = b[i];
            y = a[i];
            list1.Add(x, y);
        }
        CurveItem myCurve = myPane.AddCurve("Сукупний
операційний ризик", list1, Color.Red, SymbolType.Diamond);

        zgl.AxisChange();
    }
    if (c1.Text == "1-й інцидент")
    {
        GraphPane myPane = zgl.GraphPane;
        string l = "вирогідність";
        myPane.Title.Text = l;
        myPane.XAxis.Title.Text = "роки";
        myPane.YAxis.Title.Text = "вирогідності";
        double x, y;
        double[] a = new double[dataGridView1.RowCount];
        double[] b = new double[dataGridView1.RowCount];
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[2].Value);
            b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);

```

```

    }
    PointPairList list1 = new PointPairList();
    for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
    {
        x = b[i];
        y = a[i];
        list1.Add(x, y);
    }
    CurveItem myCurve = myPane.AddCurve("1-й
інцидент", list1, Color.Gray, SymbolType.Diamond);
    zgl.AxisChange();
}
if (c1.Text == "2-й інцидент")
{
    GraphPane myPane = zgl.GraphPane;
    string l = "виродність";
    myPane.Title.Text = l;
    myPane.XAxis.Title.Text = "роки";
    myPane.YAxis.Title.Text = "виродності";
    double x, y;
    double[] a = new double[dataGridView1.RowCount];
    double[] b = new double[dataGridView1.RowCount];
    for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
    {
        a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[3].Value);
        b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);
    }
    PointPairList list1 = new PointPairList();
    for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
    {
        x = b[i];
        y = a[i];
        list1.Add(x, y);
    }
    CurveItem myCurve = myPane.AddCurve("2-й
інцидент", list1, Color.Green, SymbolType.Diamond);
    zgl.AxisChange();
}
if (c1.Text == "3-й інцидент")
{
    GraphPane myPane = zgl.GraphPane;
    string l = "виродність";
    myPane.Title.Text = l;
    myPane.XAxis.Title.Text = "роки";
    myPane.YAxis.Title.Text = "виродності";
    double x, y;
    double[] a = new double[dataGridView1.RowCount];
    double[] b = new double[dataGridView1.RowCount];

```

```

        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[4].Value);
            b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);
        }
        PointPairList list1 = new PointPairList();
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            x = b[i];
            y = a[i];
            list1.Add(x, y);
        }
        CurveItem myCurve = myPane.AddCurve("3-й
інцидент", list1, Color.Black, SymbolType.Diamond);
        zgl.AxisChange();
    }
    if (c1.Text == "4-й інцидент")
    {
        GraphPane myPane = zgl.GraphPane;
        string l = "виродність";
        myPane.Title.Text = l;
        myPane.XAxis.Title.Text = "роки";
        myPane.YAxis.Title.Text = "виродності";
        double x, y;
        double[] a = new double[dataGridView1.RowCount];
        double[] b = new double[dataGridView1.RowCount];
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[5].Value);
            b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);
        }
        PointPairList list1 = new PointPairList();
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            x = b[i];
            y = a[i];
            list1.Add(x, y);
        }
        CurveItem myCurve = myPane.AddCurve("4-й
інцидент", list1, Color.Blue, SymbolType.Diamond);
        zgl.AxisChange();
    }
    if (c1.Text == "5-й інцидент")
    {
        GraphPane myPane = zgl.GraphPane;

```

```

        string l = "вирогідність";
        myPane.Title.Text = l;
        myPane.XAxis.Title.Text = "роки";
        myPane.YAxis.Title.Text = "вирогідності";
        double x, y;
        double[] a = new double[dataGridView1.RowCount];
        double[] b = new double[dataGridView1.RowCount];
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[6].Value);
            b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);
        }
        PointPairList list1 = new PointPairList();
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            x = b[i];
            y = a[i];
            list1.Add(x, y);
        }
        CurveItem myCurve = myPane.AddCurve("5-й
інцидент", list1, Color.Yellow, SymbolType.Diamond);
        zgl.AxisChange();
    }
    if (c1.Text == "6-й інцидент")
    {
        GraphPane myPane = zgl.GraphPane;
        string l = "вирогідність";
        myPane.Title.Text = l;
        myPane.XAxis.Title.Text = "роки";
        myPane.YAxis.Title.Text = "вирогідності";
        double x, y;
        double[] a = new double[dataGridView1.RowCount];
        double[] b = new double[dataGridView1.RowCount];
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[7].Value);
            b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);
        }
        PointPairList list1 = new PointPairList();
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
        {
            x = b[i];
            y = a[i];
            list1.Add(x, y);
        }
    }

```



```

        CurveItem myCurve = myPane.AddCurve("6-й
інцидент", list1, Color.Violet, SymbolType.Diamond);
        zgl.AxisChange();
    }
    if (c1.Text == "7-й інцидент")
    {
        GraphPane myPane = zgl.GraphPane;
        string l = "вирогідність";
        myPane.Title.Text = l;
        myPane.XAxis.Title.Text = "роки";
        myPane.YAxis.Title.Text = "вирогідності";
        double x, y;
        double[] a = new double[dataGridView1.RowCount];
        double[] b = new double[dataGridView1.RowCount];
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
            {
                a[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[8].Value);
                b[i] =
Convert.ToDouble(dataGridView1.Rows[i].Cells[0].Value);
            }
        PointPairList list1 = new PointPairList();
        for (int i = 0; i < dataGridView1.RowCount - 1;
i++)
            {
                x = b[i];
                y = a[i];
                list1.Add(x, y);
            }
        CurveItem myCurve = myPane.AddCurve("7-й
інцидент", list1, Color.Orange, SymbolType.Diamond);
        zgl.AxisChange();
    }
    }
}

```