

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра програмного забезпечення автоматизованих систем
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма «Інженерія програмного забезпечення»

«Допущений до захисту»
Завідувач кафедри

«__» _____ 2021 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти «магістр»

на тему: Нелінійна регресійна модель для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, та розробка програми для її реалізації

Виконав: студент 6151м групи

_____ Халанчук І.П.
(підпис, ПІБ)

Керівник роботи:
Викладач кафедри ПЗАС, к.т.н.
(посада, науковий ступень вчене звання)

_____ Пухалевич А.В.
(підпис, ПІБ)

Миколаїв – 2021 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
 Кафедра програмного забезпечення автоматизованих систем
 Спеціальність 121 «Інженерія програмного забезпечення»
 Освітня програма «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»

Гарант освітньої програми

_____ проф. С.Б.Приходько
 (підпис)

«___» _____ 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти «магістр»

Студенту Халанчуку Ігорю Павловичу _____
 (Прізвище, ім'я, по батькові)

1. Тема роботи: Нелінійна регресійна модель для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, та розробка програми для її реалізації

Керівник роботи Пухалевич А.В. _____

Затверджені наказом ректора № 1239уч від «13» _____ 10 _____ 2021 р.

2. Термін подання роботи: 06.12.2021 р. _____

3. Вихідні дані по роботі: _____

4. Перелік питань, що належать до розробки (найменування розділів) _____
 Вступ (Актуальність теми. Зв'язок роботи з науковими програмами, планами, темами. Мета і завдання дослідження. Об'єкт дослідження. Предмет дослідження. Методи дослідження. Наукова новизна одержаних результатів. Практичне значення одержаних результатів. Особистий внесок здобувача. Апробація результатів досліджень. Публікації.); Огляд літератури та обґрунтування необхідності проведення досліджень за обраною темою; Викладення результатів власних досліджень з висвітленням того нового, що пропонується; Проект програмного забезпечення; Організаційно-економічний розділ; Розділи з охорони праці та охорони навколишнього середовища; Висновки; Список використаних джерел; Додатки (технічне завдання, текст програми, опис програми, інструкція користувача, програма і методика випробувань програмного забезпечення)

5. Перелік презентаційних матеріалів _____

РЕФЕРАТ

Халанчук Ігор Павлович

«Нелінійна регресійна модель для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, та розробка програми для її реалізації»

Кваліфікаційна робота на здобуття освітнього рівня магістра зі спеціальності 121 – «Інженерія програмного забезпечення». Національний університет кораблебудування імені адмірала Макарова. Миколаїв, 2021 р.

Обсяг роботи: 102 стор., 4 табл., 12 рис., 24 використаних джерела, 5 додатків.

Актуальність теми роботи: необхідність підвищення достовірності оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

Мета та завдання дослідження. Метою є підвищення достовірності оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК за допомогою нелінійної регресійної моделі. Завдання дослідження: проаналізувати існуючі моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК; розробити нелінійну регресійну модель для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК; розробити програму для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

Об'єктом дослідження є процес оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

Предметом дослідження є регресійні рівняння та моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

Методи дослідження. У вирішенні поставлених завдань використано методи теорії ймовірності, математичної статистики, регресійного аналізу.

Наукова новизна одержаних результатів. Удосконалено модель ISBSG для оцінювання тривалості розробки ПЗ для ПК за рахунок, по-перше, врахування додаткової ознаки – мови програмування Visual Basic, а, по-друге, використання замість нелінійного регресійного рівняння нелінійної регресійної моделі, яка у своєму складі містить дві випадкові величини: тривалість та похибку.

Практичне значення одержаних результатів. Розроблено програму для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, для системи моделювання Scilab 6.0.0.

Апробація результатів досліджень. Результати досліджень були оприлюднені на Всеукраїнській науково-практичній інтернет-конференції.

Публікації. Результати роботи викладені у 1 публікації, а саме: 1 матеріалах Всеукраїнської науково-практичної інтернет-конференції.

Ключові слова: регресійна модель, оцінювання тривалості, програмний застосунок, Visual Basic, ПК.

ABSTRACT

Khalanchuk Igor Pavlovich

«A nonlinear regression model for estimating the development duration of applications in Visual Basic for PC and developing software for its implementation»

The qualification work in obtaining a master's degree in specialty 121 - "Software Engineering". Admiral Makarov National University of Shipbuilding. Mykolayiv, 2020.

The qualification work is presented on the 102 pages of typewritten text, contains 4 tables, 12 figures, 5 appendices and 24 references.

Relevance of the topic of the work: the need to increase the confidence of simulating the estimating the development duration of apps in Visual Basic for PC.

The goal and objectives of the study. The goal of the study is to increase the confidence of estimating the development duration of applications in Visual Basic for PC by nonlinear regression equation. The objectives of the study: to analyze existing regression equations for simulating the estimating the development duration of applications in Visual Basic for PC; to improve the regression equation for estimating the development duration of applications in Visual Basic for PC; to develop the software for estimating the development duration of apps in Visual Basic for PC.

The object of the study is the process of estimating the development duration of applications in Visual Basic for PC.

The subject of the study is the regression equations for estimating the development duration of applications in Visual Basic for PC.

The methods of the study. The methods of probability theory, mathematical statistics, regression analysis are used in solving the problems.

The scientific novelty of obtained results. The ISBSG model for estimating the development duration of applications for PC is improved based on the firstly, taking into account Visual Basic programming language, and secondly, using instead of a nonlinear regression equation a nonlinear regression model, which contains two random variables: duration and error.

The practical significance of obtained results is that the software realizing the constructed model is developed for the simulation system of Scilab 6.0.0.

Approbation of study results. The study results were presented at the All-Ukrainian scientific Internet-conference.

Publications. The study results were published in one publication: the materials of the All-Ukrainian scientific Internet-conference.

Keywords: regression model, duration estimation, software application, Visual Basic, PC.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	8
ВСТУП	9
1 АНАЛІЗ ІСНУЮЧИХ МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ VISUAL BASIC ДЛЯ ПК ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ ЗА ОБРАНОЮ ТЕМОЮ	13
1.1 Аналіз існуючих моделей для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК	13
1.2 Обґрунтування необхідності проведення досліджень за обраною темою	24
НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ VISUAL BASIC ДЛЯ ПК	26
2.1 Перевірка даних для побудови нелінійної регресійної моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, на наявність викидів	20
2.2 Побудова нелінійної регресійної моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК	23

3 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ	
ТРИВАЛОСТІ РОЗРОБКИ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ	
МОВОЮ VISUAL BASIC ДЛЯ ПК	32
3.1 Постановка задачі на розробку проекту програми для оцінювання	
тривалості розробки застосунків, що створюються мовою Visual	
Basic для ПК	32
3.2 Розробка ескізного проекту програми для оцінювання тривалості	
розробки застосунків, що створюються мовою Visual Basic для	
ПК	34
3.3 Розробка технічного проекту програми для оцінювання тривалості	
розробки застосунків, що створюються мовою Visual Basic для	
ПК	41
3.4 Розробка робочого проекту програми для оцінювання тривалості	
розробки застосунків, що створюються мовою Visual Basic для	
ПК	44
4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ	
ПРОГРАМИ	54
4.1 Розрахунок витрат на створення й експлуатацію програми	54
4.2 Економічна ефективність розробки і впровадження програми	57
5 ОХОРОНА ПРАЦІ	60
5.1 Вступ	60
5.2 Аналіз небезпечних і шкідливих факторів у офісному приміщенні з	
персональними комп'ютерами	62
5.3 Розрахунок системи штучного освітлення у офісному приміщенні з	
персональними комп'ютерами	64

5.4	Розробка заходів щодо зменшення впливу небезпечних і шкідливих факторів	67
6	ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА	70
6.1	Вступ	70
6.2	Забруднення навколишнього середовища підприємством з розробки та супроводження програмного забезпечення	73
6.3	Розробка заходів щодо зменшення забруднення	75
	ВИСНОВКИ	77
	СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	79
	ДОДАТОК А Технічне завдання на розробку програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК	83
	ДОДАТОК Б Текст програми	87
	ДОДАТОК В Опис програми	95
	ДОДАТОК Г Інструкція користувача	96
	ДОДАТОК Д Програма та методика випробувань	98

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

ПЗ – програмне забезпечення;

COCOMO – the Constructive Cost Equation;

ISBSG – International Software Benchmarking Standards Group;

KLOC – the thousand lines of code;

LB – lower bound;

LOC – lines of code;

MD – the Mahalanobis distance;

MRE – a magnitude of relative error;

MMRE – a mean magnitude of relative error;

PRED – percentage of prediction;

UB – upper bound.

ВСТУП

Актуальність теми. Проблема оцінювання тривалості розробки програмного забезпечення (ПЗ) для різних комп'ютерних платформ є важливою, оскільки, по-перше, невдала оцінка тривалості часто є основною причиною невдалих програмних проєктів [1], по-друге, вищенаведена тривалість залежить не тільки від трудомісткості [2], а і комп'ютерної платформи [3].

Незважаючи на існування відомих моделей для оцінювання тривалості розробки ПЗ, таких як COCOMO (The Constructive Cost Model) [2] та ISBSG (International Software Benchmarking Standards Group) [3], робота у напрямку побудови нових моделей для відповідного оцінювання продовжується [4-11]. Це пов'язано з низькою достовірністю оцінювання тривалості розробки ПЗ.

Як показав аналіз даних репозитарію ISBSG з релізу 2021 року, що поділ моделей тільки за комп'ютерними платформами вже є недостатнім, оскільки отримані моделі мають погані значення показників якості, таких як R^2 , PRED(0,25) та MMRE [12]. Отже, окрім комп'ютерної платформи виникає необхідність додавання інших ознак кластеризації даних. Однією з таких ознак окрім комп'ютерної платформи природньо розглядати ще і мову програмування.

На сьогодні Visual Basic є популярною мовою програмування у світі, яка за даними TIOBE Index [13] на листопад 2021 року займає шосте місце. В першу чергу ця мова застосовується для розробки програмних застосунків для платформи ПК. Існуюча модель ISBSG для оцінювання тривалості розробки ПЗ

для платформи ПК є нелінійним регресійним рівнянням, що побудовано із використанням нормалізуючого перетворення у вигляді десяткового логарифму. Ця модель не враховує те, що тривалість є випадковою величиною, розподіл якої не є гаусівським [3, 14]. Тому удосконалювати модель ISBSG ми будемо, по-перше, шляхом врахування додаткової ознаки – мови програмування Visual Basic, а, по-друге, використання замість нелінійного регресійного рівняння нелінійної регресійної моделі, яка у своєму складі містить дві випадкові величини: тривалість та похибку.

Мета і завдання дослідження. Метою роботи є підвищення достовірності оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК за допомогою нелінійної регресійної моделі.

Для досягнення поставленої мети потрібно вирішити такі завдання.

- Проаналізувати існуючі математичні моделі для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК;
- Здійснити підбор сукупності даних, за якими буде будуватися нелінійна регресійна модель для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК, та перевірити ці дані на наявність багатовимірних викидів;
- Побудувати нелінійну регресійну модель для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК на основі нормалізуючого перетворення;
- На підставі запропонованого нелінійної регресійної моделі розробити програму для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

Об'єктом дослідження є процес оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

Предметом дослідження є рівняння регресії для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

Методи дослідження. У вирішенні поставлених завдань використано методи теорії ймовірності, математичної статистики, регресійного аналізу.

Наукова новизна одержаних результатів. Удосконалено модель ISBSG для оцінювання тривалості розробки ПЗ для ПК за рахунок, по-перше, врахування додаткової ознаки – мови програмування Visual Basic, а, по-друге, використання замість нелінійного регресійного рівняння нелінійної регресійної моделі, яка у своєму складі містить дві випадкові величини: тривалість та похибку. Ця нелінійна регресійна модель в порівнянні з моделлю ISBSG, яка побудована на основі нормалізуючого перетворення у вигляді десяткового логарифму без врахування викидів у регресії, має більші значення множинного коефіцієнту детермінації і відсотка прогнозування, менші значення середньої величини відносної похибки та ширин довірчого інтервалу і інтервалу прогнозування для більшої кількості значень незалежних змінних.

Практичне значення одержаних результатів. Розроблено програму для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК. Програму створено sci-мовою для системи Scilab. Програму, що розроблено, можна використовувати для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК на ранніх стадіях проектування ПЗ.

Особистий внесок здобувача. У публікації [12], що зроблена у співавторстві, автору належать нелінійна регресійна модель для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК,

яка побудована на основі нормалізуючого перетворення у вигляді десяткового логарифму із врахуванням викидів у регресії.

Апробація результатів досліджень. Результати досліджень, викладені у роботі, були оприлюднені на II-й Всеукраїнській науково-практичній інтернет-конференції «Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2021)» (Миколаїв, 26-28 жовтня 2021 р.).

Публікації. За результатами кваліфікаційної роботи зроблено одну публікацію, а саме: матеріали Всеукраїнської науково-практичної інтернет-конференції.

1 АНАЛІЗ ІСНУЮЧИХ МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ VISUAL BASIC ДЛЯ ПК ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ ЗА ОБРАНОЮ ТЕМОЮ

1.1 Аналіз існуючих моделей для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

На сьогоднішній день існує багато математичних моделей для оцінювання тривалості розробки ПЗ, у тому числі такі відомі як COCOMO (The Constructive Cost Model) [2] та ISBSG (International Software Benchmarking Standards Group) [3]. Моделі COCOMO – це однофакторні нелінійні регресійні рівняння, які побудовані на основі нормалізуючого перетворення у вигляді десяткового логарифму в залежності від трудомісткості для трьох типів програмних застосунків: органічний, напів-розділений та вбудований. Ці три моделі мають наступний вигляд:

$$D = 0,371 \cdot E^{0,38} \text{ (органічний тип застосунків);}$$

$$D = 0,431 \cdot E^{0,35} \text{ (напів-розділений тип застосунків);}$$

$$D = 0,501 \cdot E^{0,32} \text{ (вбудований тип застосунків),}$$

де D – тривалість розробки програмного застосунку (у місяцях); E – трудомісткість розробки програмного застосунку (у людино-годинах).

Моделі ISBSG – це також однофакторні нелінійні регресійні рівняння, які побудовані на основі нормалізуючого перетворення у вигляді десяткового логарифму в залежності від трудомісткості але вже для трьох типів комп'ютерних платформ: mainframe, midrange та PC. Ці три моделі мають наступний вигляд:

$$D = 0,458 \cdot E^{0,366} \text{ для платформи mainframe;}$$

$$D = 0,548 \cdot E^{0,360} \text{ для платформи midrange;}$$

$$D = 1,936 \cdot E^{0,221} \text{ для платформи PC,} \quad (1.1)$$

де D – тривалість розробки програмного застосунку (у місяцях); E – трудомісткість розробки програмного застосунку (у людино-годинах).

Модель ISBSG для всіх типів комп'ютерних платформ має наступний вигляд:

$$D = 0,662 \cdot E^{0,328}. \quad (1.2)$$

Незважаючи на існування відомих моделей для оцінювання тривалості розробки ПЗ, таких як COCOMO та ISBSG, робота у напрямку побудови нових моделей для відповідного оцінювання продовжується [4-11]. Це пов'язано з низькою достовірністю оцінювання тривалості розробки ПЗ. Так, якщо

використати дані з 243 програмних застосунків для ПК (для платформи PC) із репозитарію ISBSG релізу 2021 року для оцінювання якості рівняння (1.1), то значення R^2 , PRED(0,25) та MMRE є відповідно такими: 0,0977, 0,2675 та 0,9877. Ці значення вказують на незадовільну якість рівняння (1.1). Кращі оцінки показників R^2 , PRED(0,25) та MMRE для зазначених даних з 243 програмних застосунків має рівняння (1.2), для якого значення R^2 , PRED(0,25) та MMRE є відповідно такими: 0,2911, 0,3498 та 0,6096. Але і вони вказують на незадовільну якість рівняння (1.2). За зазначеними даними з 243 програмних застосунків для ПК було побудовано рівняння типу (1.1), яке має вигляд

$$D = 0,4902 \cdot E^{0,3437}. \quad (1.3)$$

Хоча у порівнянні з попередніми рівняння (1.3) має дещо кращі оцінки показників R^2 та MMRE (0,2971 та 0,4840 відповідно) та приблизно таке ж значення PRED(0,25) – це 0,3457, але знову якість рівняння (1.3) є незадовільною.

Як показав аналіз даних репозитарію ISBSG з релізу 2021 року, що поділ моделей тільки за комп'ютерними платформами вже є недостатнім, оскільки отримані моделі мають погані значення показників якості, таких як R^2 , PRED(0,25) та MMRE [12]. Отже, окрім комп'ютерної платформи виникає необхідність додавання інших ознак кластеризації даних. Однією з таких ознак окрім комп'ютерної платформи природньо розглядати ще і мову програмування.

У разі однофакторного лінійної регресійної моделі

$$\hat{Y} = b_0 + b_1 X_1 \quad (1.4)$$

знаходження оцінок параметрів рівняння регресії (1.4) за методом найменших квадратів можна спростити і обчислювати за наступними формулами:

$$\hat{b}_1 = \frac{\sum_{i=1}^N (Y_i - \bar{Y})(X_{1i} - \bar{X}_1)}{\sum_{i=1}^N (X_{1i} - \bar{X}_1)^2}; \quad (1.5)$$

$$\hat{b}_0 = \bar{Y} - \hat{b}_1 \bar{X}_1, \quad (1.6)$$

$$\text{де } \bar{Y} = \frac{1}{N} \sum_{i=1}^N Y_i; \quad \bar{X}_1 = \frac{1}{N} \sum_{i=1}^N X_{1i}.$$

У разі лінійної регресії не виникає проблем і з побудовою її довірчого інтервалу та інтервалу прогнозування. Ці інтервали визначаються із використанням квантилів розподілу Стюдента. Довірчий інтервал та інтервал прогнозування та інтервал прогнозування лінійної регресії в залежності від одного фактору визначаються відповідно за наступними формулами:

$$\hat{Y}_i \pm t_{\alpha/2, N-2} S_Y \sqrt{\frac{1}{N} + \frac{(X_{1i} - \bar{X}_1)^2}{S_{X_1 X_1}}}; \quad (1.7)$$

$$\hat{Y}_i \pm t_{\alpha/2, N-2} S_Y \sqrt{1 + \frac{1}{N} + \frac{(X_{1i} - \bar{X}_1)^2}{S_{X_1 X_1}}}, \quad (1.8)$$

де $t_{\alpha/2, N-2}$ – квантиль t -розподілу Стюдента з $N-2$ ступенями свободи та $\alpha/2$

$$\text{рівнем значущості; } S_Y^2 = \frac{1}{N-2} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2; \bar{X}_1 = \frac{1}{N} \sum_{i=1}^N X_{1i}; S_{X_1 X_1} = \sum_{i=1}^N (X_{1i} - \bar{X}_1)^2.$$

У разі, якщо випадкові величини будуть мати негаусівський розподіл, рівняння регресії буде нелінійним. Але яку саме не лінійність потрібно застосовувати заздалегідь, як правило, не відомо. Тому на сьогодні для побудови нелінійної регресійної моделі в основному використовують метод простого перебору. За цим методом беруть різні не лінійності, будують сукупність нелінійних рівнянь регресії і з цієї сукупності за певним критерієм обирають найкраще рівняння. Але застосування методу простого перебору потребує багато часу і не гарантує знаходження найкращого рівняння нелінійної регресії тому, що воно в цю сукупність може не потрапити. Із-за цього для побудови нелінійної регресійної моделі прагнуть використовувати інші методи, які не потребують простого перебору. До цих методів відносять методи, що базуються на лініаризуючих та нормалізуючих перетвореннях. Методи, що використовують лінеаризацію, нажаль можуть застосовуватися лише в поодиноких випадках, коли вдається знайти лініаризуюче перетворення. І навпаки, нормалізуюче перетворення завжди вдається підібрати, що дає можливість застосовувати для побудови нелінійної регресійної моделі саме метод на основі нормалізуючих перетворень. Суть цього методу полягає у наступному. Спочатку потрібно нормалізувати емпіричні дані, для чого треба здійснити вибір нормалізуючого перетворення і оцінювання його параметрів. Далі за нормалізованими даними отримати лінійне рівняння регресії. І на останок, використовуючи обране нормалізуюче перетворення та побудоване

лінійне рівняння регресії, здійснити перехід до нелінійної регресійної моделі [15].

У разі побудови однофакторного нелінійної регресійної моделі із застосуванням одновимірного перетворення у вигляді десяткового логарифму до кожної змінної

$$Z_Y = \lg Y; \quad (1.9)$$

$$Z_1 = \lg X_1 \quad (1.10)$$

спочатку емпіричні дані нормалізують за перетвореннями (1.9) і (1.10). Далі будується однофакторне лінійне регресійне рівняння для нормалізованих даних

$$\hat{Z}_Y = b_0 + b_1 Z_1. \quad (1.11)$$

За рівнянням (1.11) та перетвореннями (1.9) і (1.10) отримують однофакторне нелінійне рівняння регресії

$$\hat{Y} = 10^{\hat{b}_0} X_1^{\hat{b}_1}. \quad (1.12)$$

Рівняння виду (1.12) з відповідними параметрами застосовуються у якості відомих моделей (таких як COCOMO та ISBSG) в інженерії програмного забезпечення: наприклад, рівняння (1.1) і (1.2).

У разі однофакторної нелінійної регресії (1.12) її довірчий інтервал та інтервал прогнозування згідно з [15] можна записати як

$$10^{\left(\hat{Z}_{Y_i} \pm t_{\alpha/2, N-2} S_Z \sqrt{\frac{1}{N} + \frac{(Z_{1_i} - \bar{Z}_1)^2}{S_{Z_1 Z_1}}} \right)}, \quad (1.13)$$

а її інтервал прогнозування – як

$$10^{\left(\hat{Z}_{Y_i} \pm t_{\alpha/2, N-2} S_Z \sqrt{1 + \frac{1}{N} + \frac{(Z_{1_i} - \bar{Z}_1)^2}{S_{Z_1 Z_1}}} \right)}, \quad (1.14)$$

де $t_{\alpha/2, N-2}$ – квантиль t -розподілу Стюдента з $N-2$ ступенями свободи та $\alpha/2$

рівнем значущості; $S_Z^2 = \frac{1}{N-2} \sum_{i=1}^N (Z_i - \hat{Z}_i)^2$; $\bar{Z}_1 = \frac{1}{N} \sum_{i=1}^N Z_{1_i}$; $S_{Z_1 Z_1} = \sum_{i=1}^N (Z_{1_i} - \bar{Z}_1)^2$.

Ми використали формули (1.13) та (1.14) для побудови довірчих інтервалів та інтервалів прогнозування нелінійних регресій виду (1.12). Так, для нелінійного регресійного рівняння (1.3) інтервал прогнозування для 0,05 рівня значущості наведений на рис. 1.1. На рис. 1.1 на осі абсцис відкладено трудомісткість (у людино-годинах) розробки програмних застосунків для ПК, а на осі ординат – тривалість розробки (у місяцях). На рис. 1.1 відповідна регресія показана суцільною лінією чорного кольору, а нижня та верхня границі її інтервалу прогнозування – суцільними лініями червоного кольору. Також на рис. 1.1 наведені 243 точки даних з репозитарію ISBSG релізу 2021 року, за якими будувалися нелінійна регресія (1.3) та нижня і верхня границі її інтервалу прогнозування.

Як можна побачити з рис. 1.1, є точки даних, що виходять за границі інтервалу прогнозування нелінійної регресії (1.3). Таких точок виявилось 13, які згідно з [16, 17] були відкинуті, і ми знову побудували нелінійне регресійне

рівняння (1.12) але вже для 230 точок даних. Оцінки параметрів рівняння (1.12) для 230 точок даних є такими: $\hat{b}_0 = -0,25524$ та $\hat{b}_1 = 0,32804$.

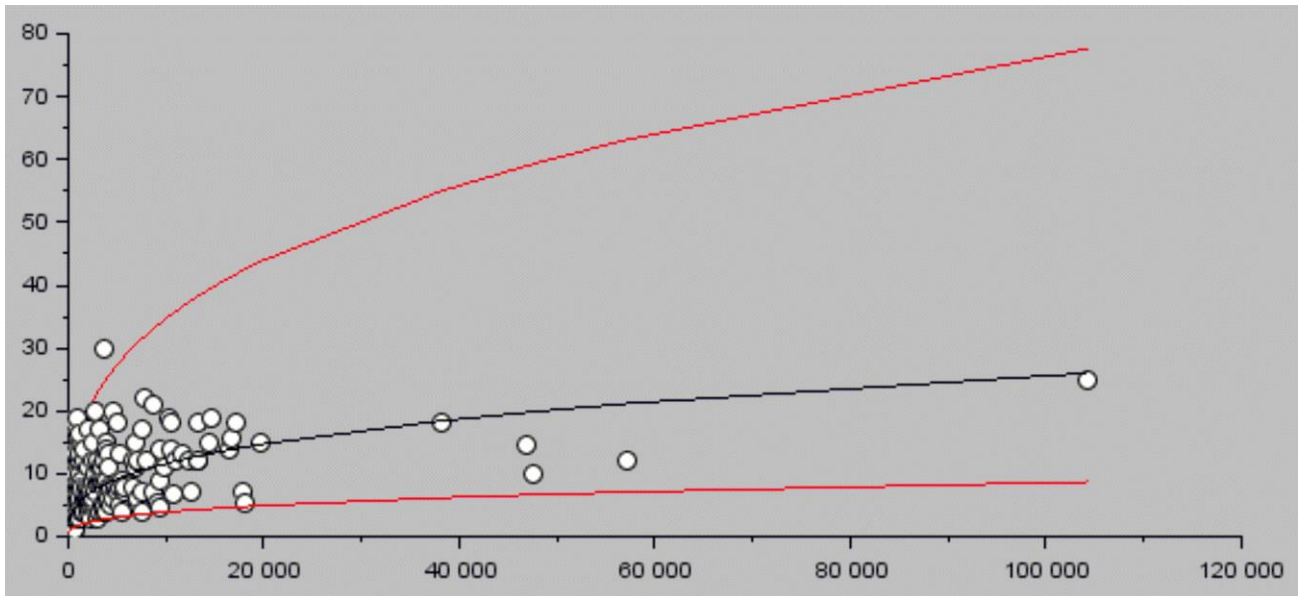


Рисунок 1.1 – Нелінійна регресія (1.3) та її інтервал прогнозування

У порівнянні з рівнянням (1.3) з оцінками параметрів для 243 точок даних, рівнянням (1.12) з оцінками параметрів для 230 точок даних має дещо кращі оцінки показників R^2 , $PRED(0,25)$ та $MMRE$ (0,3761, 0,3609 та 0,4131 відповідно). Але якість рівняння (1.12) з оцінками параметрів для 230 точок даних також є незадовільною.

Ми використали формулу (1.14) для побудови інтервалу прогнозування нелінійної регресії (1.12) з оцінками параметрів для 230 точок даних для 0,05 рівня значущості. Цей інтервал прогнозування для 0,05 рівня значущості наведений на рис. 1.2. На рис. 1.12 на осі абсцис відкладено трудомісткість (у людино-годинах) розробки програмних застосунків для ПК, а на осі ординат – тривалість розробки (у місяцях). На рис. 1.2 відповідна регресії показана

суцільною лінією чорного кольору, а нижня та верхня границі її інтервалу прогнозування – суцільними лініями червоного кольору. Також на рис. 1.2 наведені 230 точок даних з репозитарію ISBSG релізу 2021 року, за якими будувалися нелінійна регресія (1.12) та нижня і верхня границі її інтервалу прогнозування.

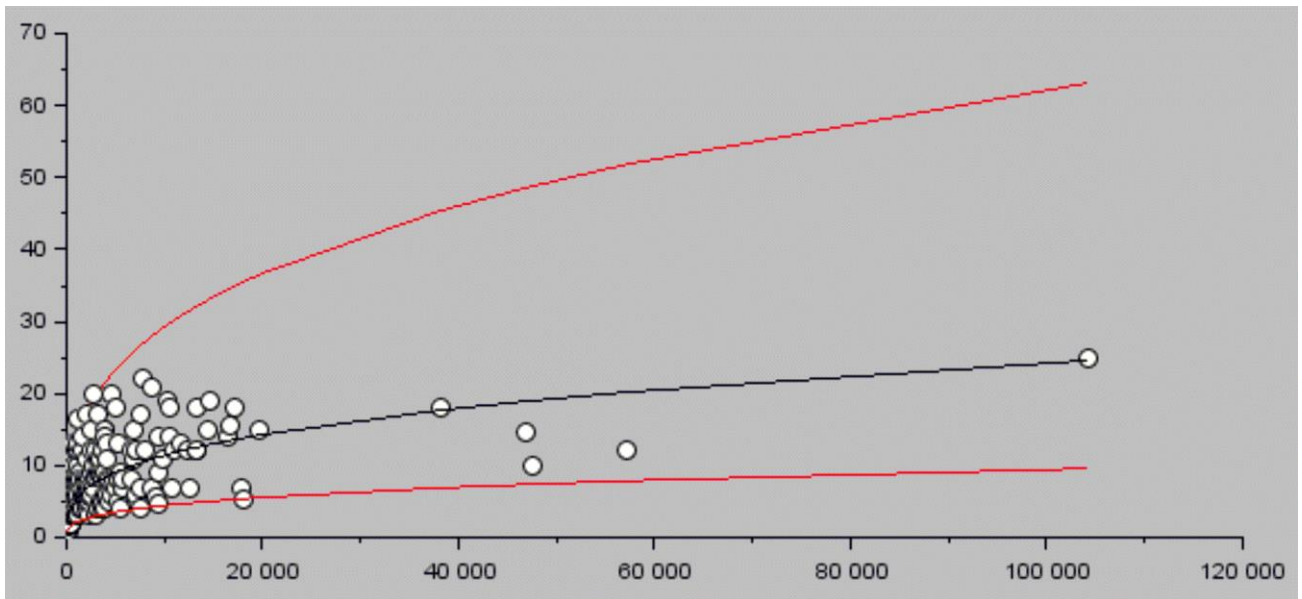


Рисунок 1.2 – Нелінійна регресія (1.12) та її інтервал прогнозування для 230 точок даних

Як можна побачити з рис. 1.2, є точки даних, що виходять за границі інтервалу прогнозування нелінійної регресії (1.12). Таких точок виявилось 7, які згідно з [16, 17] були відкинуті, і ми знову побудували нелінійне регресійне рівняння (1.12) але вже для 223 точок даних. Оцінки параметрів рівняння (1.12) для 223 точок даних є такими: $\hat{b}_0 = -0,27177$ та $\hat{b}_1 = 0,33251$.

У порівнянні з рівнянням (1.12) з оцінками параметрів для 230 точок даних, рівнянням (1.12) з оцінками параметрів для 223 точок даних має кращі

оцінки показників R^2 , PRED(0,25) та MMRE (0,4405, 0,3767 та 0,3895 відповідно). Але знову якість рівняння (1.12) з оцінками параметрів для 223 точок даних є незадовільною.

Ми використали формулу (1.14) для побудови інтервалу прогнозування нелінійної регресії (1.12) з оцінками параметрів для 223 точок даних для 0,05 рівня значущості. Виявилося 4 точки даних, які згідно з [16, 17] були відкинуті, і ми знову продовжили процес побудови нелінійного регресійного рівняння (1.12) але вже для 219 точок даних. Всього було 11 таких ітерацій, після яких залишилося 204 точки даних. У порівнянні з рівнянням (1.12) з оцінками параметрів, що були знайдені на попередніх ітераціях, рівнянням (1.12) з оцінками параметрів для 204 точок даних має кращі оцінки показників R^2 , PRED(0,25) та MMRE (0,5306, 0,4118 та 0,3342 відповідно). Але знову якість рівняння (1.12) з оцінками параметрів для 204 точок даних є незадовільною.

Ми знову використали формулу (1.14) для побудови інтервалу прогнозування нелінійної регресії (1.12) з оцінками параметрів вже для 204 точок даних для 0,05 рівня значущості. Цей інтервал прогнозування для 0,05 рівня значущості наведений на рис. 1.3. На рис. 1.3 на осі абсцис відкладено трудомісткість (у людино-годинах) розробки програмних застосунків для ПК, а на осі ординат – тривалість розробки (у місяцях). На рис. 1.3 відповідна регресія показана суцільною лінією чорного кольору, а нижня та верхня границі її інтервалу прогнозування – суцільними лініями червоного кольору. Також на рис. 1.3 наведені 204 точки даних з репозитарію ISBSG релізу 2021 року, за якими будувалися нелінійна регресія (1.12) та нижня і верхня границі її інтервалу прогнозування.

Як можна побачити з рис. 1.3, немає точок даних, що виходять за границі інтервалу прогнозування нелінійної регресії (1.12). Тому згідно з [16, 17] ми

припинили процес побудови нелінійного регресійного рівняння (1.12). Остаточно оцінки параметрів рівняння (1.12) для 204 точок даних є такими: $\hat{b}_0 = -0,41649$ та $\hat{b}_1 = 0,37120$.

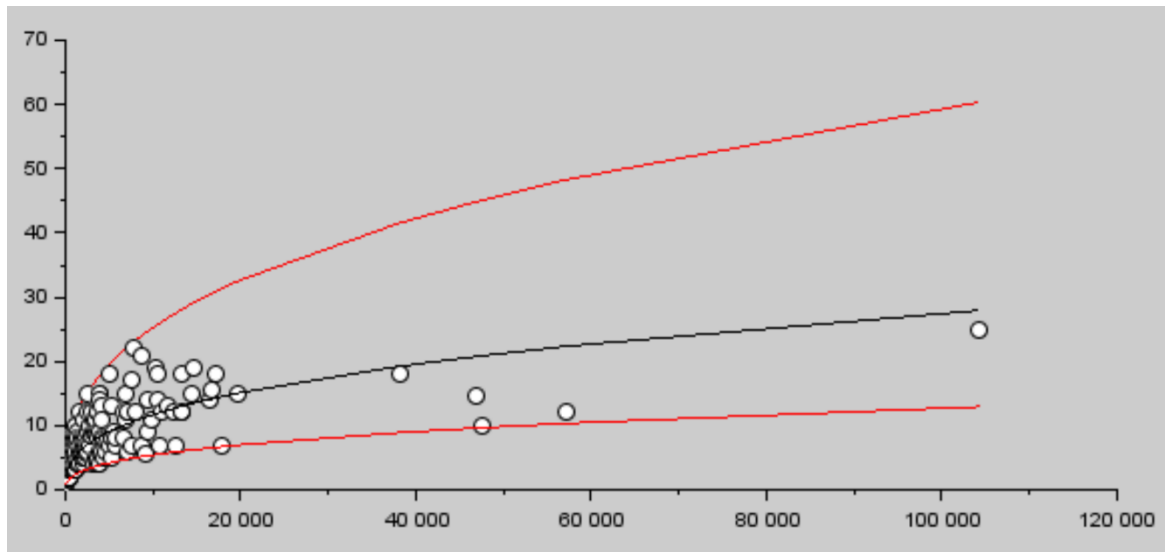


Рисунок 1.3 – Нелінійна регресія (1.12) та її інтервал прогнозування для 204 точок даних

Отже, як показує аналіз даних репозитарію ISBSG з релізу 2021 року, поділ моделей тільки за комп'ютерними платформами вже є недостатнім, оскільки отримані моделі мають погані значення показників якості, таких як R^2 , $PRED(0,25)$ та $MMRE$. Тому окрім комп'ютерної платформи виникає необхідність додавання інших ознак кластеризації даних. Однією з таких ознак окрім комп'ютерної платформи природньо розглядати ще і мову програмування.

Практичне застосування нормалізуючих перетворень для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, потребує пошуку відповідного перетворення. У якості нормалізуючого

перетворення зазвичай використовується перетворення у вигляді десяткового логарифму.

1.2 Обґрунтування необхідності проведення досліджень за обраною темою

Як було зазначено в попередньому підрозділі, проблема оцінювання тривалості ПЗ для різних комп'ютерних платформ є важливою, оскільки, по-перше, невдала оцінка тривалості часто є основною причиною невдалих програмних проєктів [1], по-друге, вищенаведена тривалість залежить не тільки від трудомісткості [2], а і комп'ютерної платформи [3].

Незважаючи на існування відомих моделей для оцінювання тривалості розробки ПЗ, таких як COCOMO [2] та ISBSG [3], робота у напрямку побудови нових моделей для відповідного оцінювання продовжується [4-11]. Це пов'язано з низькою достовірністю оцінювання тривалості розробки ПЗ.

Як показав аналіз даних репозитарію ISBSG з релізу 2021 року, поділ моделей тільки за комп'ютерними платформами вже є недостатнім, оскільки отримані моделі мають погані значення показників якості, таких як R^2 , PRED(0,25) та MMRE. Тому окрім комп'ютерної платформи виникає необхідність додавання інших ознак кластеризації даних. Однією з таких ознак окрім комп'ютерної платформи природньо розглядати ще і мову програмування.

На сьогодні Visual Basic є популярною мовою програмування у світі, яка за даними TIOBE Index [13] на листопад 2021 року займає шосте місце. В першу чергу ця мова застосовується для розробки програмних застосунків для

платформи ПК. Існуюча модель ISBSG для оцінювання тривалості розробки ПЗ для платформи ПК є нелінійним регресійним рівнянням, що побудовано із використанням нормалізуючого перетворення у вигляді десяткового логарифму. Ця модель не враховує те, що тривалість є випадковою величиною, розподіл якої не є гаусівським [3, 14]. Тому удосконалювати модель ISBSG ми будемо, по-перше, шляхом врахування додаткової ознаки – мови програмування Visual Basic, а, по-друге, використання замість нелінійного регресійного рівняння нелінійної регресійної моделі, яка у своєму складі містить дві випадкові величини: тривалість та похибку.

Таким чином, виникає потреба у проведенні досліджень за обраною темою.

НЕЛІНІЙНА РЕГРЕСІЙНА МОДЕЛЬ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ VISUAL BASIC ДЛЯ ПК

2.1 Перевірка даних для побудови нелінійної регресійної моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК на наявність викидів

Для побудови нелінійної регресійної моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, використовується вибірка двовимірних негаусових даних з репозитарію ISBSG з релізу 2021 року: тривалість розробки застосунку у місяцях Y та трудомісткість розробки у людино-годинах X_1 . Ці дані для 39 застосунків наведені на рис. 2.1.

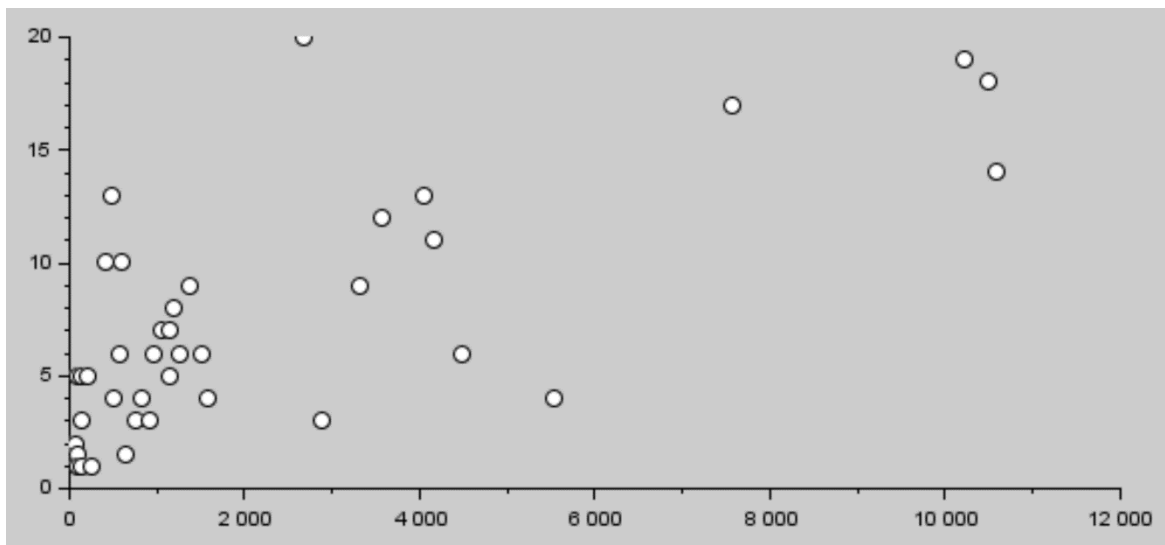


Рисунок 2.1 – Тривалість розробки 39 застосунків (у місяцях) в залежності від трудомісткості (у людино-годинах)

Для виявлення викидів у цих даних ми використовуємо метод, оснований на нормалізуючих перетвореннях і квадраті відстані Махаланобіса (Mahalanobis Distance) [19, 20], яку далі позначаємо як MD. Квадрат MD для кожної багатовимірної точки даних i , $i = 1, 2, \dots, N$, визначається як

$$d_i^2 = (\mathbf{Z}_i - \bar{\mathbf{Z}})^T S_N^{-1} (\mathbf{Z}_i - \bar{\mathbf{Z}}), \quad (2.1)$$

де $\bar{\mathbf{Z}}$ – вектор вибірових середніх; S_N – вибіркова коваріаційна матриця

$$S_N = \frac{1}{N} \sum_{i=1}^N (\mathbf{Z}_i - \bar{\mathbf{Z}})(\mathbf{Z}_i - \bar{\mathbf{Z}})^T. \quad (2.2)$$

Тут $\mathbf{Z}_i$ – гаусівський випадковий вектор $\mathbf{Z} = \{Z_1, Z_2, \dots, Z_m\}^T$ для кожної багатовимірної точки даних i , $i = 1, 2, \dots, N$; m – кількість компонентів вектору $\mathbf{Z}$. В нашому випадку m дорівнює 2.

Вектор $\mathbf{Z}$ отримують шляхом перетворення негаусівського випадкового вектору $\mathbf{X} = \{X_1, X_2, \dots, X_m\}^T$ за допомогою багатовимірного нормалізуючого перетворення

$$\mathbf{Z} = \psi(\mathbf{X}). \quad (2.3)$$

В нашому випадку в перетворенні (2.3) $\psi = \{\lg Y, \lg X_1\}^T$.

Відомо [11], якщо дані мають багатовимірний нормальний розподіл, то розподіл квадрату MD поводитья як розподіл χ^2 . Для великих $N - m$ (принаймні більше 25), відстань d_i^2 повинна вести себе приблизно як незалежні випадкові величини $\chi_{m,\alpha}^2$. Тут $\chi_{m,\alpha}^2$ – це квантиль розподілу χ^2 з рівнем значимості α . Значення даних, для яких величина квадрату MD більша, ніж квантиль розподілу χ^2 , розглядаються як викиди, і ці значення відкидаються. Після відкидання зменшене число багатомірних точок даних знову

нормалізується, використовуючи перетворення (2.3), доки всі значення квадрату MD (2.1) будуть меншими чи рівними квантилю розподілу χ^2 .

Для рівня значимості 0,005 та перетворення у вигляді десяткового логарифму викиди відсутні тому, що для всіх рядків даних квадрат MD є меншим ніж величина квантіля розподілу χ^2 , яка становить 10,60 для рівня значимості 0,005 та двох ступенів свободи.

В подальшому двовимірні негаусові дані, що наведені на рис. 2.1, використовуються для побудови нелінійної регресійної моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

2.2 Побудова нелінійної регресійної моделі для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Для побудови нелінійної регресійної моделі для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК, спочатку двовимірні негаусівські дані з рис. 2.1 ми нормалізуємо за допомогою одномірного перетворення у вигляді десяткового логарифму. Далі для нормалізованих даних будемо лінійну регресійну модель

$$Z_Y = \hat{Z}_Y + \varepsilon = \hat{b}_0 + \hat{b}_1 Z_1 + \varepsilon, \quad (2.4)$$

де $Z_Y = \lg Y$; $Z_1 = \lg X_1$, ε – це гаусівська випадкова величина, яка визначає залишки, $\varepsilon \sim N(0, \sigma_\varepsilon^2)$.

Параметри лінійної моделі (2.4) оцінювалися методом найменших

квадратів. Оцінки параметрів такі: $\hat{b}_0 = -0,45133$ та $\hat{b}_1 = 0,39827$. Оцінка середньо квадратичного відхилення $\hat{\sigma}_\varepsilon$ дорівнює 0,25827.

Після цього будуємо нелінійну регресійну модель за (2.4) та перетворенням у вигляді десяткового логарифму

$$Y = 10^{\hat{b}_0 + \varepsilon} X_1^{\hat{b}_1}. \quad (2.5)$$

Далі використовуємо формулу (1.14) для побудови інтервалу прогнозування нелінійної регресії (2.5) з оцінками параметрів для 39 точок даних для 0,05 рівня значущості. Цей інтервал прогнозування для 0,05 рівня значущості наведений на рис. 2.2. На рис. 2.2 на осі абсцис відкладено трудомісткість (у людино-годинах) розробки програмних застосунків для ПК, а на осі ординат – тривалість розробки (у місяцях). На рис. 2.2 відповідна регресії показана суцільною лінією чорного кольору, а нижня та верхня границі її інтервалу прогнозування – суцільними лініями червоного кольору. Також на рис. 2.2 наведені 39 точок даних з репозитарію ISBSG релізу 2021 року, за якими будувалися нелінійна регресійна модель (2.5) та нижня і верхня границі інтервалу прогнозування регресії.

Як можна побачити з рис. 2.2, немає точок даних, що виходять за границі інтервалу прогнозування нелінійної регресії (2.5). Тому згідно з [16, 17] ми припинили процес побудови нелінійної регресійної моделі (2.5). Остаточні оцінки параметрів нелінійної регресійної моделі (2.5) для 39 точок даних є такими: $\hat{b}_0 = -0,45133$ та $\hat{b}_1 = 0,39827$. Оцінка середньо квадратичного відхилення $\hat{\sigma}_\varepsilon$ дорівнює 0,25827.

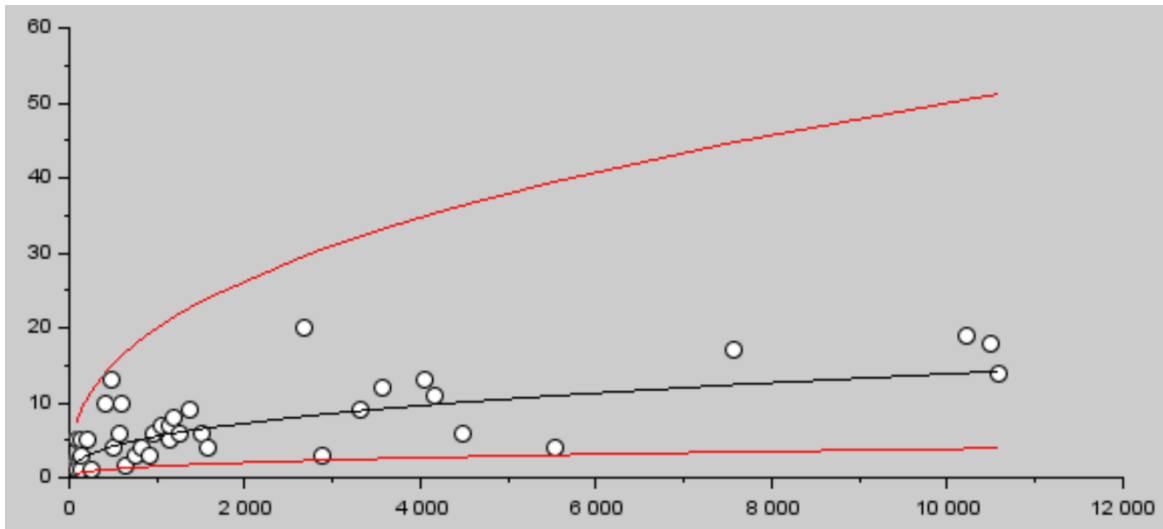


Рисунок 2.2 – Нелінійна регресія тривалості розробки 39 застосунків (у місяцях) в залежності від трудомісткості (у людино-годинах) та її інтервал прогнозування

Математична модель (2.5) – це і є нелінійна регресійна модель для оцінювання тривалості розробки застосунків (у місяцях), що створюються мовою Visual Basic для ПК, в залежності від трудомісткості їх розробки (у людино-годинах).

Нелінійна регресійна модель (2.5) є кращою за рівняння (1.12), яке побудовано за 243 точками даних з репозитарію ISBSG релізу 2021 року для застосунків, що створювалися для ПК. Нелінійна регресійна модель (2.5) має кращі значення показників R^2 , PRED(0,25) та MMRE (0,4824, 0,3590 та 0,5372 відповідно) порівняно з рівнянням (1.12), яке побудовано за 243 точками даних з репозитарію ISBSG релізу 2021 року для застосунків, що створювалися для ПК, для якого значення показників R^2 , PRED(0,25) та MMRE (0,4018, 0,3333 та

0,5450 відповідно) визначалися для тих самих 39 точок даних, що і для моделі (2.5).

Нелінійна регресійна модель (2.5) є кращою за рівняння (1.12), яке побудовано і за 204 точками даних з репозитарію ISBSG релізу 2021 року для застосунків, що створювалися для ПК. Нелінійна регресійна модель (2.5) має кращі значення показників R^2 та PRED(0,25) порівняно з рівнянням (1.12), яке побудовано за 204 точками даних з репозитарію ISBSG релізу 2021 року для застосунків, що створювалися для ПК, для якого значення показників R^2 та PRED(0,25) (0,3840 та 0,2821 відповідно) визначалися для тих самих 39 точок даних, що і для моделі (2.5). При цьому значення MMRE для моделі (2.5) на 3,8% гірше за відповідне значення для рівняння (1.12).

Отримані результати свідчать про те, що для побудови регресійних рівнянь, в тому числі і для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, потрібно використовувати нормалізуючі перетворення з урахуванням викидів у регресії.

3 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ РОЗРОБКИ ЗАСТОСУНКІВ, ЩО СТВОРЮЮТЬСЯ МОВОЮ VISUAL BASIC ДЛЯ ПК

3.1 Постановка задачі на розробку проекту програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Виходячи з аналізу існуючих методів та рівнянь для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, сформулюємо постановку задачі. Потрібно створити програму для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, яка би задовольняла таким вимогам.

Вимоги до програми.

Вимоги функціональних характеристик.

Вимоги до складу функцій, що виконуються.

Програма має виконувати такі функції:

1) Вводити з текстового файлу емпіричні дані з метрик програмних застосунків, що створені мовою Visual Basic для ПК (значення випадкових величин): тривалість розробки застосунків у місяцях Y та трудомісткість розробки у людино-годинах X_1 .

2) Вводити з інтерфейсу програми довірчу ймовірність, трудомісткість (у людино-годинах) розробки програмних застосунків, що створюються мовою

Visual Basic для ПК, за якими потрібно оцінити тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.

3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням у вигляді десяткового логарифму.

4) Знаходити значення коефіцієнтів лінійної регресійної моделі за нормалізованими емпіричними даними.

5) Здійснювати побудову нелінійної регресійної моделі.

6) Обчислювати значення коефіцієнту детермінації R^2 .

7) Здійснювати побудову інтервалу прогнозування нелінійної регресії.

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії тривалості (у місяцях) розробки програмних застосунків, що створюються мовою Visual Basic для ПК, значення коефіцієнту детермінації R^2 .

9) Виводити у текстовий файл результати оцінювання тривалості (у місяцях) розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

Вимоги до складу й параметрів технічних засобів.

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2, оперативна пам'ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Вимоги до інформаційної та програмної сумісності.

Для запуску та роботи з програмним забезпеченням необхідна операційна система Windows Vista/7/8/10 (32 або 64 біти), система моделювання

Scilab 6.0.0., компілятор C (Visual Studio 2010 або Visual Express 2010) для компіляції зовнішніх модулів C або C++ та використання моделі в Xcos.

На підставі цієї постановки задачі розроблено технічне завдання, яке наведено у додатку А.

3.2 Розробка ескізного проекту програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

В ескізному проекті було виявлено варіанти використання та побудовано їх сценарії, розроблено зовнішню специфікацію програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, та стани цієї програми при виконанні.

Даний проект передбачає наявність лише одного актора – користувача програми, який буде повноцінно використовувати всі функції програми. Виявлені варіанти використання знаходяться в таблиці 3.1.

Таблиця 3.1 Виявлені варіанти використання

Основні актори	Найменування	Формулювання
1	2	3
Користувач	Ввести дані метрик ПЗ	Вводить дані про фактичну тривалість розробки застосунку у місяцях та трудомісткість розробки у людино-годинах.

Продовження таблиці 3.1

1	2	3
Користувач	Ввести трудомісткість розробки ПЗ	Вводить трудомісткість (у людино-годинах) розробки програмного застосунку, що створюється мовою Visual Basic для ПК, за яким потрібно оцінити тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.
Користувач	Ввести довірчу ймовірність	Вводить довірчу ймовірність, для якої потрібно оцінити довірчий інтервал та інтервал прогнозування нелінійної регресії тривалості (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.
Користувач	Нормалізувати дані	Виконує нормалізацію внесених даних
Користувач	Побудувати нелінійну регресію	Будує нелінійну регресію для оцінювання середнього значення тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК
Користувач	Отримати інтервали нелінійної регресії	Отримує довірчий інтервал та інтервал прогнозування нелінійної регресії тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК, із заданою довірчою ймовірністю

Діаграма варіантів використання програми для оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК, зображена на рисунку 3.1.

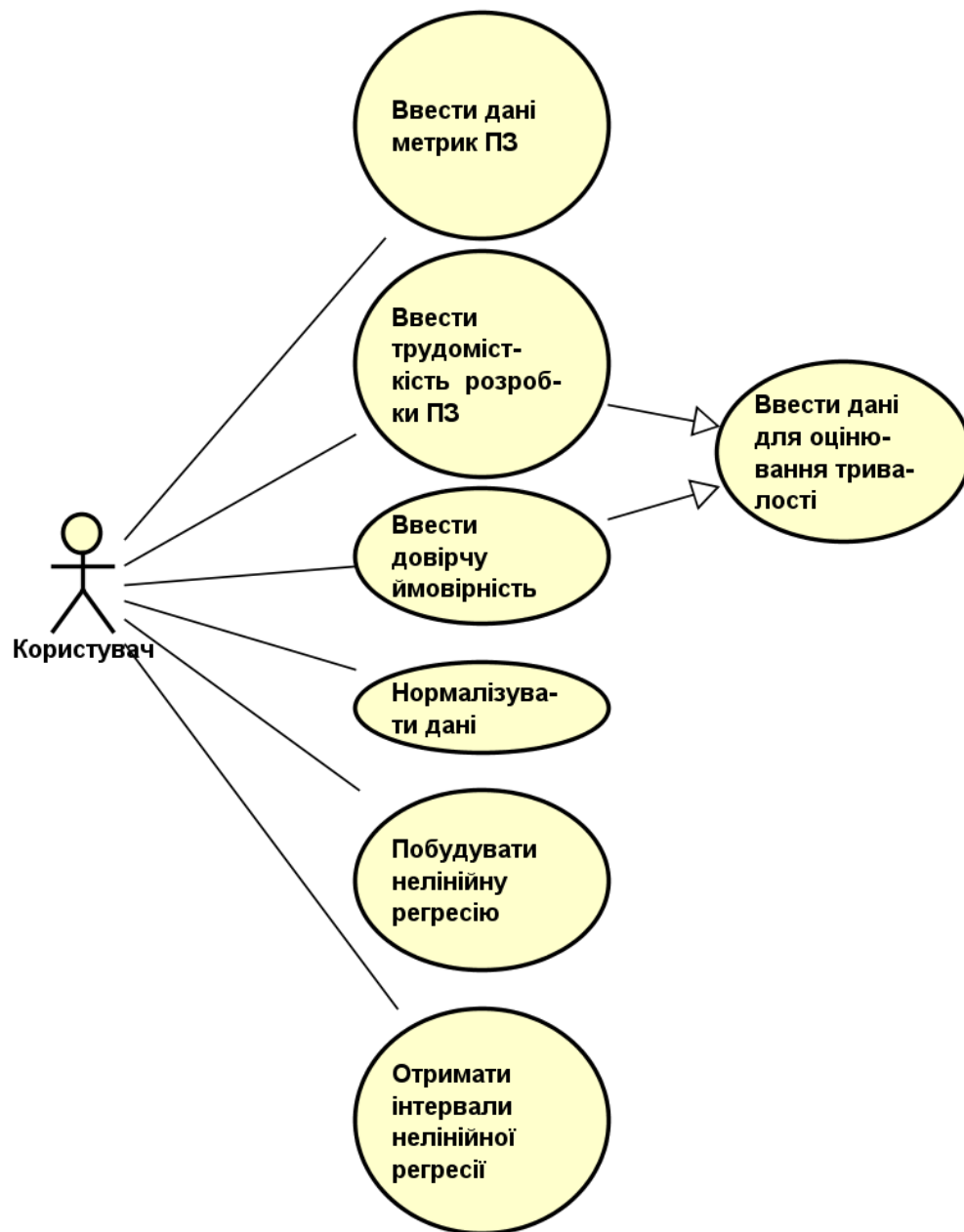


Рисунок 3.1 – Діаграма варіантів використання ПЗ для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Сценарії варіантів використання зображені у вигляді діаграми діяльності на рисунку 3.2.

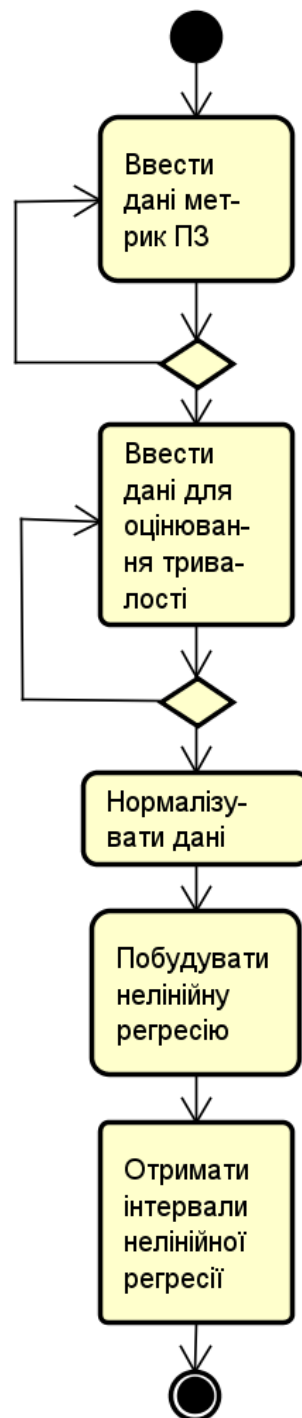


Рисунок 3.2 – Діаграма діяльності ПЗ для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Зовнішня специфікація програми.

Назва програми: VB_AppDurEst.

Функції, які виконує програма:

1) Вводити з текстового файлу емпіричні дані з метрик програмних застосунків, що створені мовою Visual Basic для ПК (значення випадкових величин): перша метрика Y включає тривалість розробки застосунків у місяцях та друга метрика X_1 визначає трудомісткість розробки у людино-годинах.

2) Вводити з інтерфейсу програми довірчу ймовірність та трудомісткість (у людино-годинах) розробки програмних застосунків, що створюються мовою Visual Basic для ПК, за якими потрібно оцінити тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.

3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням у вигляді десяткового логарифму.

4) Знаходити значення коефіцієнтів лінійної регресійної моделі за нормалізованими емпіричними даними.

5) Здійснювати побудову нелінійної регресійної моделі.

6) Обчислювати значення коефіцієнту детермінації R^2 .

7) Здійснювати побудову довірчого інтервалу та інтервалу прогнозування нелінійної регресії.

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії тривалості (у місяцях) розробки програмних застосунків, що створюються мовою Visual Basic для ПК, значення коефіцієнту детермінації R^2 .

9) Виводити у текстовий файл результати оцінювання тривалості розробки програмних застосунків у місяцях.

Вхідні дані:

- емпіричні дані з метрик програмних застосунків, що створені мовою Visual Basic для ПК (значення випадкових величин): перша метрика Y включає тривалість розробки застосунків у місяцях та друга метрика X_1 визначає трудомісткість розробки у людино-годинах;

- значення довірчої ймовірності та трудомісткість розробки у людино-годинах, за якою потрібно оцінити тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.

Вихідні дані:

- значення нелінійної регресії та границь довірчого інтервалу і інтервалу прогнозування нелінійної регресії тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

- значення коефіцієнту детермінації.

Зовнішні ефекти: поява графічного вікна для вводу даних, управління виводом даних, графічного вікна для виводу даних, нелінійної регресії та довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, вихід з програми (закриття всіх графічних вікон).

Для проектування інтерфейсу програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК представимо стани програми при виконанні у вигляді відповідної діаграми станів. Діаграма станів програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК наведена на рисунку 3.3.

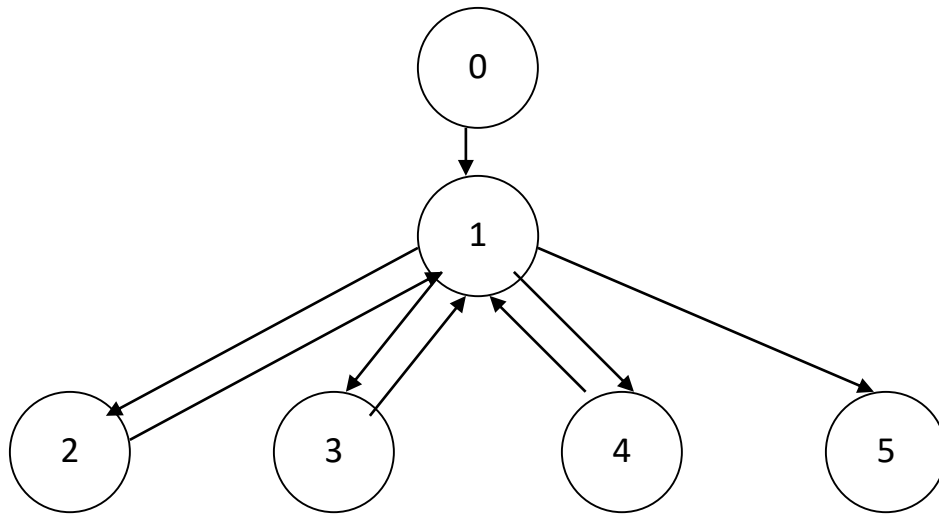


Рисунок 3.3 – Діаграма станів програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

На діаграмі станів програми введені такі позначення станів: 0 – початковий стан, 1 – графічне вікно головного модуля, 2 – графічне вікно візуалізації введених емпіричних даних, 3 – графічне вікно із результатами знаходження нелінійної регресії, 4 – графічне вікно з результатами побудови довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, 5 – кінцевий стан.

3.3 Розробка технічного проекту програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

В технічному проекті було спроектовано програму для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, на основі покрокової деталізації зверху вниз, розроблено специфікації модулів програми, побудовано алгоритми модулів програми.

Програма складається з таких модулів: головного модуля (VB_AppDurEst), модуля для введення емпіричних даних з файлу та їх візуалізації (DataInput), модуля для побудови нелінійної регресії та довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК (VB_AppDurIntEst), модуля для закриття графічних вікон (W_Close).

Специфікація головного модуля.

Назва модуля: VB_AppDurEst.

Функції, які виконує програма:

- 1) Відкриття графічного вікна головного модуля.
- 2) Введення імені файлу з емпіричними даними.
- 3) Виклик функції для введення емпіричних даних з файлу та їх візуалізації у графічному вікні.
- 4) Вибір наявності фактичного розміру ПЗ та введення його значення у тисячах строк коду.
- 5) Виклик функції для визначення нелінійної регресії та довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.
- 6) Виклик функції для закриття графічних вікон.

Вхідні дані:

- емпіричні дані з метрик програмних застосунків, що створені мовою Visual Basic для ПК (значення випадкових величин): перша метрика Y включає тривалість розробки застосунків у місяцях та друга метрика X_1 визначає трудомісткість розробки у людино-годинах;

- значення довірчої ймовірності та трудомісткість розробки у людино-годинах, за якою потрібно оцінити тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.

Вихідні дані:

- значення нелінійної регресії та границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

- значення коефіцієнту детермінації.

Зовнішні ефекти: поява графічного вікна для вводу даних, управління виводом даних, побудови довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, вихід з програми (закриття всіх графічних вікон).

Специфікація модуля для введення емпіричних даних з файлу та їх візуалізації.

Назва модуля: DataInput.

Функції, які виконує модуль:

- 1) Введення емпіричних даних (значень випадкових величини) з файлу.
- 2) Формування одномірних та двомірних масивів значень випадкових величини (емпіричних даних).

3) Відкриття графічного вікна для візуалізації введених емпіричних даних.

4) Візуалізацію введених емпіричних даних.

Вхідні дані:

- значення випадкових величини (емпіричні дані).

Вихідні дані:

- одномірні та двомірні масиви значення випадкових величини (емпіричні дані);

- кількість значень випадкових величин (емпіричних даних);

- ідентифікатор графічного вікна.

Зовнішні ефекти: Поява графічного вікна для візуалізації введених емпіричних даних.

Специфікація модуля для визначення нелінійної регресії та довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

Назва модуля: VB_AppDurIntEst.

Функції, які виконує модуль:

1) Визначає нелінійну регресію та довірчий інтервал та інтервал прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

2) Обчислює значення коефіцієнту детермінації для нелінійної регресії.

Вхідні дані:

- двомірні масиви значень випадкових величини (емпіричних даних);

- кількість значень випадкових величин (емпіричних даних).

Вихідні дані:

- значення нелінійної регресії, нижньої та верхньої границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК;

- значення коефіцієнту детермінації для нелінійної регресії.

Зовнішні ефекти: з'являються результати нелінійної регресії, нижньої та верхньої границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, та виводиться значення коефіцієнту детермінації для нелінійної регресії.

Назва модуля: W_Close.

Функції, які виконує модуль:

Закриття графічних вікон.

Вхідні дані:

- ідентифікатори графічних вікон.

Вихідні дані: відсутні.

Зовнішні ефекти: Закриття всіх графічних вікон.

В подальшому були розроблені алгоритми зазначених модулів.

3.4 Розробка робочого проекту програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

В робочому проекті було розроблено програму для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК,

здійснено тестування програми, розроблено програмну документацію, здійснено випробування програми.

Спочатку sci-мовою системи моделювання Scilab було виконано кодування алгоритмів, що були розроблені на стадії технічного проекту. Текст програми наведений в додатку Б. Приклад кодування модуля VB_AppDurIntEst, що здійснює оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, sci-мовою в системі моделювання Scilab наведено нижче.

```
//Функція оцінювання тривалості розробки застосунків,
//що створюються мовою Visual Basic для ПК
function VB_AppDurIntEst(x, yx, n)
//Считываем значение переменных из текстовых полей и
//преобразовываем их к числовому типу.
x1i=eval(get(edit_text4,'string'));
//x2i=eval(get(edit_text5,'string'));
//x3i=eval(get(edit_text6,'string'));
Pconf=eval(get(edit_text7,'string'));

x1A=0; x2A=0; x3A=0; yA=0;
for i=1:n
    x1A=x1A+x(i,1);
    //x2A=x2A+x(i,2); x3A=x3A+x(i,3);
    yA=yA+yx(i,1);
end;
x1A=x1A/n;
//x2A=x2A/n; x3A=x3A/n;
yA=yA/n;
Sx1x1=0; Sx2x2=0; Sx3x3=0;
Sx1x2=0; Sx1x3=0; Sx2x3=0; SyA=0;
for i=1:n
    Sx1x1=Sx1x1+(x(i,1)-x1A)*(x(i,1)-x1A);
    //Sx1x2=Sx1x2+(x(i,1)-x1A)*(x(i,2)-x2A);
    //Sx1x3=Sx1x3+(x(i,1)-x1A)*(x(i,3)-x3A);
    //Sx2x3=Sx2x3+(x(i,2)-x2A)*(x(i,3)-x3A);
    //Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    //Sx3x3=Sx3x3+(x(i,3)-x3A)*(x(i,3)-x3A);
    SyA=SyA+(yx(i,1)-yA)*(yx(i,1)-yA);
end;
//Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2; Sxx(1,3)=Sx1x3;
```

```

//Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2; Sxx(2,3)=Sx2x3;
//Sxx(3,1)=Sx1x3; Sxx(3,2)=Sx2x3; Sxx(3,3)=Sx3x3;

//Sxx_1=1/Sxx;

z1A=0; z2A=0; z3A=0; zYA=0;
for i=1:n
    z1A=z1A+log10(x(i,1));
    //z2A=z2A+log10(x(i,2)); z3A=z3A+log10(x(i,3));
    zYA=zYA+log10(yx(i,1));
end;
z1A=z1A/n;
//z2A=z2A/n; z3A=z3A/n;
zYA=zYA/n;
Sz1z1=0; Sz2z2=0; Sz3z3=0;
Sz1z2=0; Sz1z3=0; Sz2z3=0;
SZYz1=0;
for i=1:n
    SZYz1=SZYz1+(log10(yx(i,1))-zYA)*(log10(x(i,1))-z1A);
    Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    //Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    //Sz1z3=Sz1z3+(log10(x(i,1))-z1A)*(log10(x(i,3))-z3A);
    //Sz2z3=Sz2z3+(log10(x(i,2))-z2A)*(log10(x(i,3))-z3A);
    //Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
    //Sz3z3=Sz3z3+(log10(x(i,3))-z3A)*(log10(x(i,3))-z3A);
end;
//Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2; Szz(1,3)=Sz1z3;
//Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2; Szz(2,3)=Sz2z3;
//Szz(3,1)=Sz1z3; Szz(3,2)=Sz2z3; Szz(3,3)=Sz3z3;

//Szz_1=1/Szz;

//zi(1)=log10(x(1,1))-z1A; zi(2)=log10(x(1,2))-z2A;
zi(3)=log10(x(1,3))-z3A;
//ziT=zi';
//ziSzz_1=ziT*Szz_1;
//result=ziSzz_1*zi;
zi_res=log10(x(1,1))-z1A;
result=zi_res*zi_res/Sz1z1;

Df=n-2;
P=0.975;
Q=1-P;

[T]=cdft("T",Df,P,Q);

```

```

b1=SzYz1/Sz1z1;
b0=zYA-b1*z1A;

Szyzyn4=0; Syy=0;
for i=1:n
    Szyzyn4=Szyzyn4+...
        (b0+b1*log10(x(i,1))-log10(yx(i,1)))^2;

    Syy=Syy+(yx(i,1)-10^b0*x(i,1)^b1)^2;
end;
Szyzyn4=Szyzyn4/(n-2)
Delta_z=T*(Szyzyn4*(1/n+result))^0.5;
Delta_z_pr=T*(Szyzyn4*(1+1/n+result))^0.5;

r_2=1-Syy/SyA;

zy_result=b0+b1*log10(x1i);

zyLBC=zy_result-Delta_z; zyUBC=zy_result+Delta_z;

zyLBPr=zy_result-Delta_z_pr; zyUBPr=zy_result+Delta_z_pr;

y_result=10^zy_result;
yLBC=10^zyLBC; yUBC=10^zyUBC;
yLBPr=10^zyLBPr; yUBPr=10^zyUBPr;

file_PC5=fopen('D:\Data_F\VB_Duration_39_data5.txt','w+');
//for i=1:3
    fprintf(file_PC5, "%5.6f ", yUBC);
//end
fclose(file_PC5);

f2=figure(); //Создание графического окна.
//Устанавливаем размеры окна.
set(f2,'position',[50,50,700,300]);
//Устанавливаем заголовок окна.
set(f2,'figure_name','Регресія тривалості (у місяцях) та її
інтервали прогнозування в залежності від трудомісткості (у людино-
годинах)');

for i=1:n
    y(i)=yx(i,1);
    x1(i)=x(i,1);
    zr(i)=b0+b1*log10(x(i,1));

```

```

    yr(i)=10^zr(i);
    zi_res=log10(x(1,1))-z1A;
    result=zi_res*zi_res/Sz1z1;
    Delta_z=T*(Szyzyn4*(1+1/n+result))^0.5;

    pizrmin=zr(i)-Delta_z;
    pizrmax=zr(i)+Delta_z;

    piyr_min(i)=10^pizrmin;
    piyr_max(i)=10^pizrmax;

end;

plot2d(x1,piyr_max,5);
plot2d(x1,y,-9);
plot2d(x1,yr,1);
plot2d(x1,piyr_min,5);

set(textresult,'string', sprintf("=%1.4f",y_result));
set(textresult2,'string', sprintf("=%1.4f",yLBC));
set(textresult3,'string', sprintf("=%1.4f",yUBC));
set(textresult4,'string',...
    sprintf("Коефіцієнт детермінації R^2=%1.4f",r_2));
set(textresult5,'string', sprintf("=%1.4f",yLBPr));
set(textresult6,'string', sprintf("=%1.4f",yUBPr));

if (get(radius_bikv_x,'value')==0)

else
    yfact=eval(get(edit_text12,'string'));
    MRE=abs((yfact-y_result)/yfact);

textresult5=uicontrol(f,'style','text','string','', 'position',...
    [500,120,160,20]);
    set(textresult5,'string',...
        sprintf("Відносна похибка MRE=%1.4f",MRE));
end;

endfunction

```

Далі в середовищі Scilab 6.0.0 було створено ссе-файл розміром 15 КБ. Зазначимо, що в програмі використовувався динамічний спосіб створення інтерфейсних компонентів. Він полягає у тому, що при виконанні програми

можуть створюватися і знищуватися ті чи інші елементи управління (командні кнопки, радіокнопки, мітки та таке інше), а їх властивостям присвоюються відповідні значення. В програмі були використані такі інтерфейсні компоненти: вікно редагування, командні кнопки та радіокнопки. Графічне вікно програми із зазначеними інтерфейсними компонентами наведено на рис.3.4.

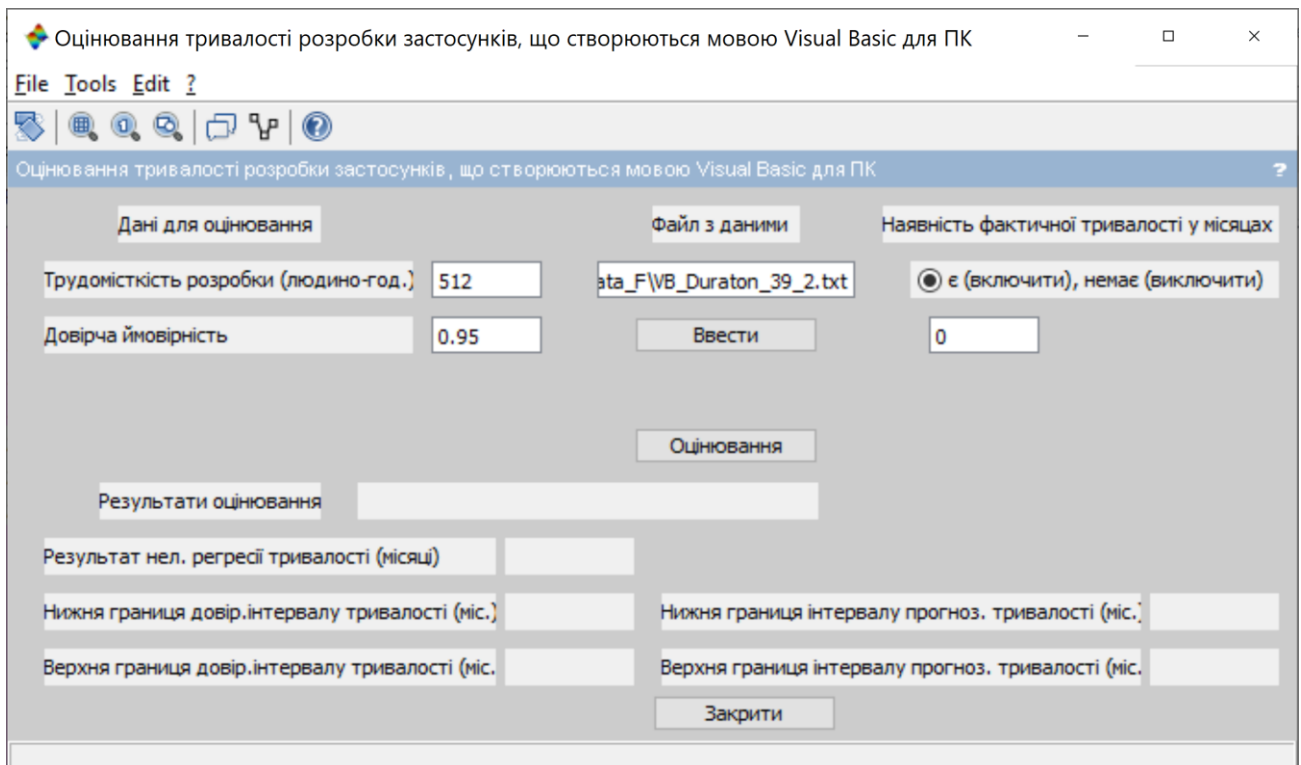


Рисунок 3.4 – Графічне вікно програми із інтерфейсними компонентами

Було проведено тестування окремих модулів. У якості прикладу наведемо результати тестування для модуля оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, для емпіричних даних – тривалості (у місяцях) розробки 39 програмних застосунків в залежності від трудомісткості їх розробки (у людино-годинах). Було побудовано нелінійну регресійну модель

на основі нормалізуючого перетворення у вигляді десяткового логарифму. Результати візуалізації емпіричних даних про тривалість (у місяцях) розробки 39 програмних застосунків в залежності від трудомісткості їх розробки (у людино-годинах) наведені на рис.3.5.

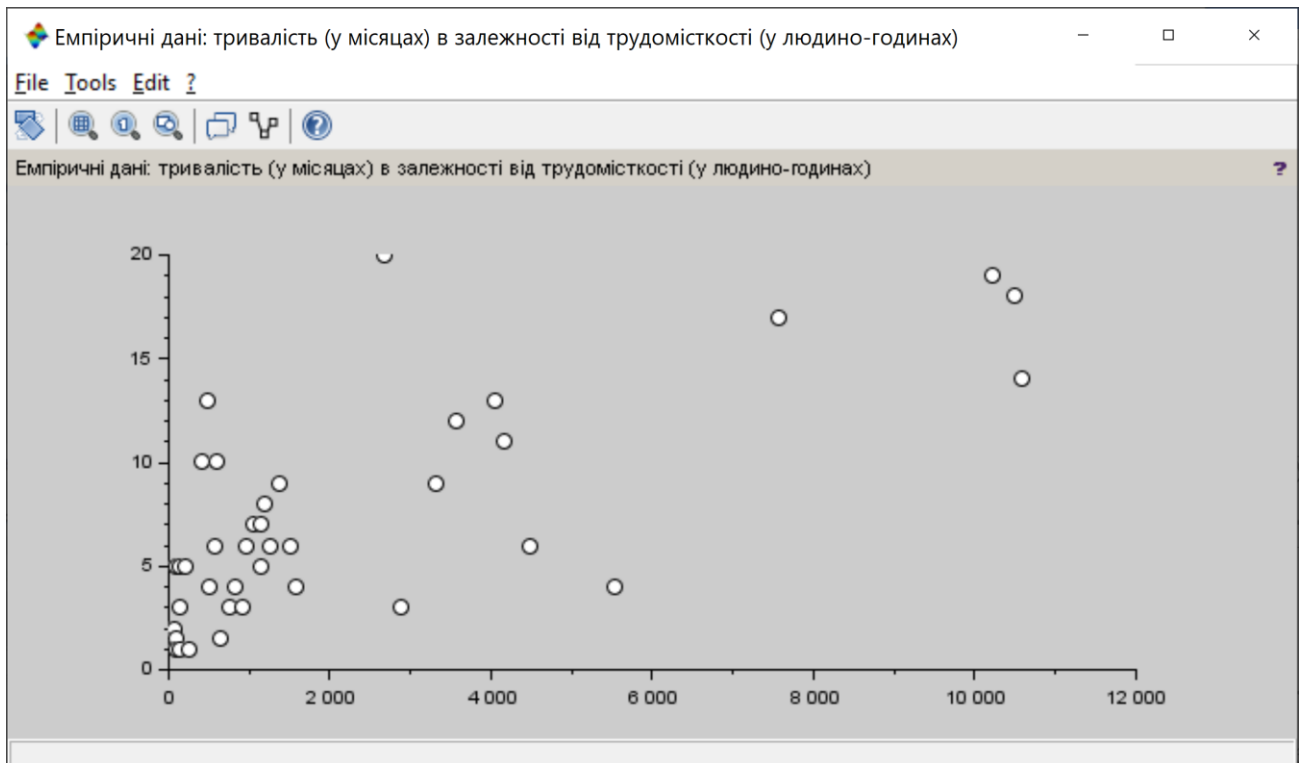


Рисунок 3.5 – Емпіричні дані про тривалість (у місяцях) розробки 39 програмних застосунків в залежності від трудомісткості їх розробки (у людино-годинах)

Результати оцінювання значень нелінійної регресії, нижньої і верхньої границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК,

та результати обчислення коефіцієнту детермінації R^2 наведені на рис.3.6. На рис.3.6 також наведено значення відносної похибки (MRE) результату оцінювання значень нелінійної регресії в порівнянні з фактичним значенням тривалості розробки застосунків, що створюються мовою Visual Basic для ПК. Фактична тривалість складає 2 місяці. Відносна похибка результату оцінювання дорівнює 1,29%.

Дані для оцінювання		Файл з даними	Наявність фактичної тривалості у місяцях
Трудомісткість розробки (людино-год.)	80	ata_F\VB_Duraton_39_2.txt	☒ є (включити), ☐ немає (виключити)
Довірча ймовірність	0.95	Ввести	2

Оцінювання

Результати оцінювання	
Коефіцієнт детермінації $R^2=0.4824$	Відносна похибка $MRE=0.0129$
Результат нел. регресії тривалості (місяці)	=2.0259
Нижня границя довір. інтервалу тривалості (міс.)	=1.3650
Верхня границя довір. інтервалу тривалості (міс.)	=3.0068
Нижня границя інтервалу прогноз. тривалості (міс.)	=0.5614
Верхня границя інтервалу прогноз. тривалості (міс.)	=7.3109

Закрити

Рисунок 3.6 – Результати оцінювання значень нелінійної регресії, нижньої і верхньої границь довірчого інтервалу та інтервалу прогнозування нелінійної регресії тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

На рис.3.7 наведені результати візуалізації нелінійної регресії (суцільна лінія чорного кольору), нижньої і верхньої границь інтервалу прогнозування (суцільні лінії червоного кольору) нелінійної регресії тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК. На рис.3.7 також нанесені 39 точок даних, за якими було побудовано нелінійну регресію, нижню і верхню границі інтервалу прогнозування для 0,05 рівня значущості.

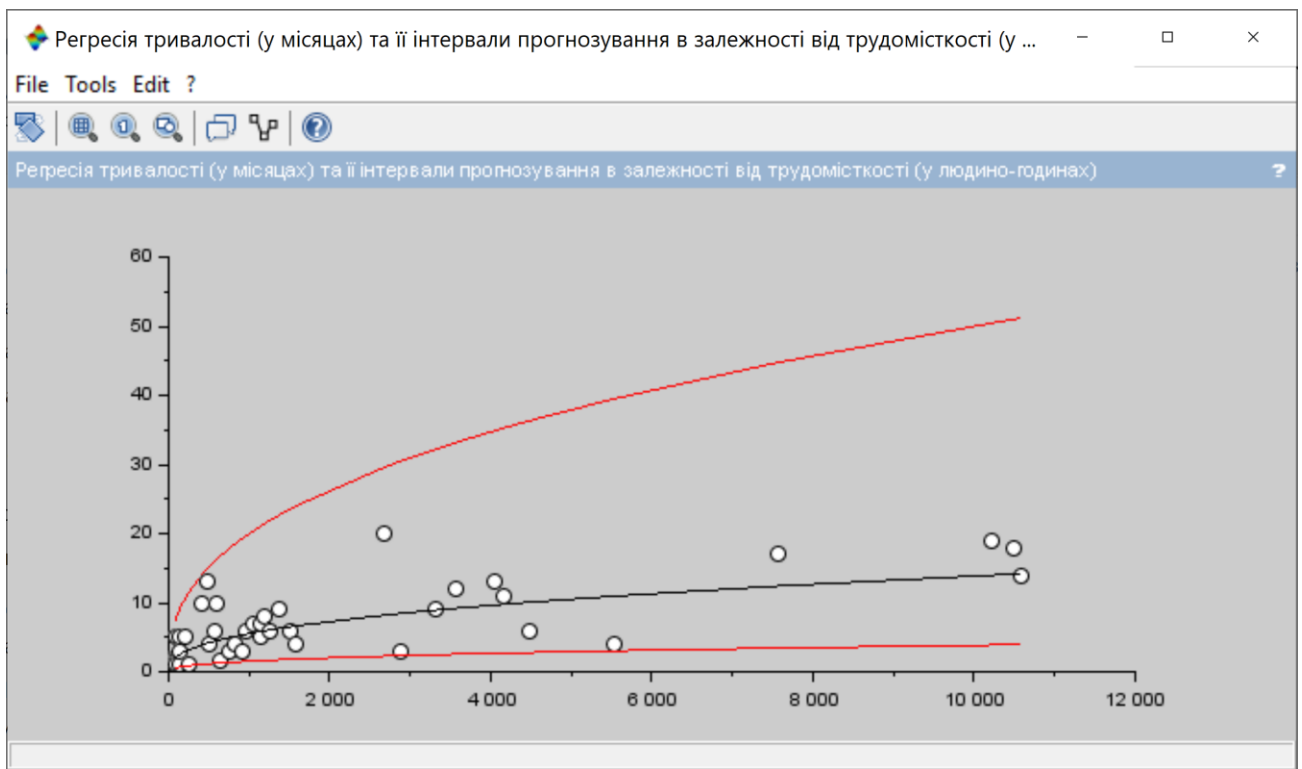


Рисунок 3.7 – Нелінійна регресія, нижня і верхня границі інтервалу прогнозування нелінійної регресії тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК

Отримані результати практично співпали з результатами, що отримані іншими авторами [12]. Це свідчить про працездатність розробленої програми.

4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ ПРОГРАМИ

Найбільш важливим моментом при проектуванні комп'ютерної програми для розробника, з економічної точки зору, є процес формування її вартості. Вочевидь, що комп'ютерна програма являє собою дуже специфічний товар з безліччю особливостей.

Створення програми вимагає одноразових витрат на її розробку, придбання необхідних технічних засобів і поточних витрат на функціонування продукту. Економія від функціонування програми визначається з урахуванням витрат на її експлуатацію. Відношення цієї економії до витрат на створення програми характеризує економічну ефективність капітальних вкладень. Економічні показники визначаються по діючим на момент розрахунку оптовим цінам, тарифам і ставкам заробітної плати.

4.1 Розрахунок витрат на створення й експлуатацію програми

Витрати на розробку програми складаються з витрат на зарплату розробника, на амортизацію комп'ютера, на якому виконується розробка, на експлуатацію цього комп'ютера, на засоби розробки та витрат на матеріали і комплектуючі.

Розробка програми виконується програмістом, місячний оклад якого складає 15000 грн. Додаткова заробітна плата складає 20% від основної. Виходячи з цього, основна і додаткова заробітна плата програміста

3000 грн/мес, а вартість сучасного ПК складає 16800 грн. (приведена вартість комп'ютера на базі процесора Intel Celeron Dual-Core J1800 (2.41 ГГц) / RAM 4 ГБ / HDD 500 ГБ / Intel HD Graphics / LAN / Windows 8 SL та монітора 18.5" Asus VS197DE). При вартості кіловат-години електроенергії у 1,7166 грн. (II клас напруги до 27,5 кВт, разом з ПДВ), розраховується вартість розробки ПЗ. Витрати на допоміжні матеріали приведені в табл. 4.1

Таблиця 4.1 – Витрати на допоміжні матеріали

Пункти витрат	Сума
Папір	90,00
Заправлення картриджа до принтера	90,00
CD-диск	6,00
Непередбачені витрати	150,00
Разом матеріали і комплектуючі	336,00

Вартість програми розрахуємо за формулою:

$$\text{Спр} = (\text{Ззп} + \text{Зсз} + \text{Ззг} + \text{Зе}) * \text{T} + \text{Зм}, \quad (4.1)$$

де Т – тривалість розробки, міс.; Ззп – основна і додаткова заробітна плата обслуговуючого персоналу, грн.; Зсз – відрахування на соціальні заходи або єдиний соціальний внесок (36,76% від основної і додаткової заробітної плати), грн.; Ззг – загальногосподарські витрати (10% від основної заробітної плати), грн.; Зе – витрати на електроенергію; Зм – витрати на основні і допоміжні матеріали. Зе при споживанні потужності 0,5 кВт, тривалості роботи за місяць, рівної $22 * 8 = 176$ годин, вартості кіловат-години електроенергії 1,7166 грн. складає

$$\text{Зе} = 176 * 1,7166 * 0,5 = 151,06 \text{ грн.}$$

Таблиця 4.2 – Витрати на розробку ПЗ

Найменування витрат	Одиниця	Кількість
Тривалість розробки	Мес	1,5
Основна і додаткова ЗП	Грн.	18000
Відрахування на соціальні заходи	Грн.	6616,8
Загальногосподарські витрати	Грн.	250
Витрати на матеріали	Грн.	336,00
Витрати на електроенергію	Грн.	151,06

Відповідно до формули (4.1) вартість програми:

$$\text{Спр} = (18000 + 6616,8 + 250 + 151,06) * 1,5 + 336,00 = 25353,86 \text{ грн.}$$

Амортизаційні відрахування на устаткування складають 60% балансової вартості в рік:

$$\text{Аоб} = 16800 * 0,6 = 10080 \text{ грн.}$$

У масштабах підприємства річні витрати на основні і допоміжні матеріали (гнучкі диски, папір) визначаються в розмірі 5% вартості основного устаткування:

$$\text{Вм} = 16800 * 0,05 = 840 \text{ грн.}$$

Річний обсяг робіт ПК у годинах визначається в такий спосіб:

$$\Phi_{\text{м}} = 264,5 * T_{\text{з}}, \quad (4.2)$$

де $T_{\text{з}}$ – це середнє місячне завантаження устаткування (близько 4 годин); 264,5 – середня кількість робочих днів у році.

Отже, річний обсяг роботи ПК складе:

$$\Phi_{\text{м}} = 264,5 * 4 = 1058 \text{ годин.}$$

Витрати на електроенергію Z_e при 1013 годинах роботи устаткування в рік складуть

$$Z_e = 1058 * 0,5 * 1,7166 = 908,08 \text{ грн.}$$

Експлуатаційні витрати для ПК за рік складуть:

$$Z_{зр} = 10080 + 840 + 908,08 = 11828,08 \text{ грн.}$$

Отже, у перший рік витрати на створення й експлуатацію програми складуть:

$$Z_{се} = 25353,86 + 11828,08 = 37181,94 \text{ грн.}$$

4.2 Економічна ефективність розробки і впровадження програми

Основним показником економічної ефективності функціонування програми є підвищення ефективності керування інформацією у вигляді зниження витрат на керування при одночасному збільшенні швидкості і якості одержання потрібного результату.

Крім багатьох інших негативних ефектів, ручна обробка інформації спричиняє наступні негативні економічні ефекти:

- високі витрати на складування паперових документів (сейфи, шафи, папки й ін.);
- підвищені витрати на канцтовари;
- витрати, пов'язані з роботою виявлення раніше допущених помилок (людський фактор).

До числа основних факторів, що визначають приріст прибутку в зв'язку з упровадженням програми, відносяться:

- підвищення продуктивності праці;
- вивільнення робочого часу.

Крім того, не піддається прямій грошовій оцінці підвищення оперативності керування, якість одержуваних результатів, поліпшення організації праці і т.д.

Обов'язковою умовою визначення економічної ефективності програми є порівнянність усіх показників у часі, за цінами й іншими нормами, використовуваними для визначення показників, за змістом і колом елементів витрат.

Визначимо пряму економічну ефективність, ґрунтуючись на тому, що впровадження програмного продукту вивільняє 0,6 працівника (за експертною оцінкою фахівців підприємства).

Зарплата 0,6 працівника в рік складає:

$$(15000 \cdot 12) \cdot 0,6 = 108000 \text{ грн.}$$

Річний економічний ефект розраховується по формулі:

$$E_p = \Delta C_n - E_n \cdot k,$$

де ΔC_n – вивільнені кошти після впровадження програми (108000 грн.) мінус експлуатаційні витрати (11828,08 грн.); E_n – коефіцієнт ефективності (дорівнює коефіцієнту амортизації – 0,6); k – одноразові витрати на впровадження програми (25353,86 грн.).

Річний економічний ефект буде

$$E_p = 96171,92 - 25353,86 \cdot 0,6 = 80959,60 \text{ грн.}$$

Строк окупності програми розраховується по формулі:

$$T = \frac{k}{\Delta C_n} = \frac{25353,86}{96171,92} = 0,26 \text{ (року)}.$$

Отже строк окупності програми складає приблизно 3,2 місяця.

5 ОХОРОНА ПРАЦІ

5.1 Вступ

Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

Державна політика в галузі охорони праці базується на принципах:

1) пріоритету життя і здоров'я працівників відповідно до результатів виробничої діяльності підприємства і, повної відповідальності власника за створення безпечних і нешкідливих умов праці;

2) комплексного розв'язання завдань охорони праці на основі національних програм з цих питань та з урахуванням інших напрямів економічної і соціальної політики, досягнень, в галузі науки і техніки та охорони навколишнього середовища;

3) соціального захисту працівників, повного відшкодування шкоди особам, які потерпіли від нещасних випадків на виробництві і професійних захворювань;

4) встановлення єдиних нормативів з охорони праці для всіх підприємств, незалежно від форм власності і видів їх діяльності;

5) використання економічних методів управління охороною праці, проведення політики пільгового оподаткування, що сприяє створенню

безпечних і нешкідливих умов праці, участі держави у фінансуванні заходів щодо охорони праці;

6) здійснення навчання населення, професійної підготовки і підвищення кваліфікації працівників з питань охорони праці;

7) забезпечення координації діяльності державних органів, установ, організацій та громадських об'єднань, що вирішують різні проблеми охорони здоров'я, гігієни та безпеки праці, а також співробітництва і проведення консультацій між власниками та працівниками (їх представниками);

8) соціальний захист працівників, повне відшкодування шкоди особам, потерпілим від нещасних випадків на виробництві та професійних захворювань;

9) установлення єдиних нормативів з охорони праці для всіх підприємств незалежно від форм власності і видів їх діяльності;

10) використання економічних методів управління охороною праці, проведення політики пільгового, оподаткування, що сприяє створенню безпечних та нешкідливих умов праці, участь держави в фінансуванні заходів з охорони праці;

11) здійснення навчання населення, професійної підготовки та підвищення кваліфікації працівників з питань охорони праці;

12) забезпечення координації діяльності державних органів, установ, організацій та об'єднань громадян, що вирішують різні проблеми охорони здоров'я, гігієни та безпеки праці, а також співробітництво.

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а

також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці.

Власник або уповноважений ним орган повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизму, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників.

На власника або уповноважений ним орган покладається систематичне проведення інструктажу (навчання) працівників з питань охорони праці, протипожежної охорони.

Охорона праці – один з центральних інститутів трудового права. Він має виключно практичне значення. Недодержання вимог охорони праці створює небезпеку для здоров'я і життя працівників. У свою чергу і ті, кого законодавець називає власником або уповноваженим ним органом, несуть сувору, у тому числі й кримінальну відповідальність за порушення правил охорони праці.

5.2 Аналіз небезпечних і шкідливих факторів у офісному приміщенні з персональними комп'ютерами

Працівники, що працюють у офісному приміщенні з персональними комп'ютерами (ПК) піддаються шкідливій дії ПК, тому робочі місця користувачів повинні відповідати безпечним і нешкідливим умовам праці.

У зв'язку з цим передбачається розробити комплекс мір, що забезпечують безпечні і нешкідливі умови праці і розглянути екологічні питання.

При виконанні посадових обов'язків на працівників у офісному приміщенні, що працюють з ПК постійно або періодично діють наступні небезпечні і шкідливі фактори:

- 1) забруднення повітря шкідливими речовинами, пилом, мікроорганізмами;
- 2) невідповідність нормам параметрів мікроклімату;
- 3) виникнення на екрані монітора електростатичних зарядів, що змушують часточки пилу рухатися до найближчого заземленого предмету, часто нею виявляється особа користувача;
- 4) підвищений рівень шуму на робочому місці;
- 5) підвищений рівень статичної електрики при невірно спроектованій робочій зоні;
- 6) небезпечний рівень напруги в електричному ланцюзі, замикання якого може відбутися через тіло людини;
- 7) широкий спектр випромінювання від дисплея, що включає рентгенівську ультрафіолетову й інфрачервону області, а так само широкий діапазон електромагнітних випромінювань інших частот;
- 8) підвищений рівень електромагнітних випромінювань;
- 9) підвищений рівень іонізуючих випромінювань(м'яке рентгенівське та гамма-випромінювання);
- 10) відсутність або недолік природного світла;
- 11) недостатня освітленість робочої зони;
- 12) підвищена яскравість світла;
- 13) знижена контрастність;
- 14) пряма і зворотна блескність;
- 15) підвищена пульсація світлового потоку (мерехтіння зображення);

- 16) тривале перебування в тому самому положенні, і повторення тих самих рухів приводить до синдрому тривалих статичних навантажень (СТСН);
- 17) нераціональна організація робочого місця;
- 18) невідповідність ергономічних характеристик устаткування нормованим величинам;
- 19) розумова перенапруга, що обумовлена характером розв'язуваних задач приводить до синдрому тривалих психологічних навантажень (СТПН);
- 20) великий обсяг інформації, що переробляється, приводить до значних навантажень на органи зору та нервову систему;
- 21) монотонність праці;
- 22) нервово-психічні навантаження;
- 23) нервово-емоційні стресові навантаження;
- 24) небезпека виникнення пожежі.

5.3 Розрахунок системи штучного освітлення у офісному приміщенні з персональними комп'ютерами

У приміщенні необхідно організувати змішане освітлення, тобто сполучення природного і штучного освітлення. Природне освітлення створюється бічним освітленням через вікно. Штучне освітлення використовується при недостатньому природному освітленні. У даному приміщенні використовується загальне штучне освітлення.

Розрахунок його здійснюється по методу світлового потоку з урахуванням потоку, відбитого від стін і стелі.

Потрібно зробити розрахунок загального освітлення для офісного приміщення. Довжина $A=4,5$ м, ширина $B=3,0$ м, висота $H=2,5$ м. Висота робочої поверхні $h_p = 1,0$ м. Для освітлення використовується світильник УПМ-15. Мінімальна освітленість лампи накаливання по нормах $E_{\min}=75$ лк. Коефіцієнт відображення стелі $S_p=70\%$, стін $S_c=50\%$, робочої поверхні $S_r=10\%$. Напруга мережі 220 В.

Відстань від стелі до робочої поверхні:

$$H_o = H - h_p = 2,5 - 1 = 1,5 \text{ м.}$$

Відстань від стелі до світильника:

$$h_c = 0,2 * H_o = 0,2 * 1,5 = 0,3 \text{ м.}$$

Висота підвісу світильника над освітленою поверхнею:

$$h = H_o - h_c = 1,5 - 0,3 = 1,2 \text{ м.}$$

Висота підвісу світильника над підлогою:

$$H_n = h + h_p = 1,2 + 1 = 2,2 \text{ м.}$$

Для досягнення найбільшої рівномірності освітлення приймаємо відношення $L/h = 1,5$.

Відстань між центрами світильників:

$$L = 1,5 * h = 1,5 * 1,2 = 1,8 \text{ м.}$$

Необхідна кількість світильників:

$$N = S/L^2 = (4,5 * 3) / (1,8)^2 = 4,2 \approx 4.$$

Приймаємо кількість світильників: 4.

Індекс приміщення:

$$i = (A * B) / (h * (A + B)) = (4,5 * 3) / (1,2 * (4,5 + 3)) = 1,5.$$

Коефіцієнт використання світлового потоку $\eta = 0,51$.

Світловий потік однієї лампи:

$$\begin{aligned} \Phi &= (E_{\min} * S * K_3 * Z) / (N * \eta) = \\ &= (75 * 13,5 * 1,3 * 1,15) / (4 * 0,51) = 741 \text{ лм.} \end{aligned}$$

За знайденим світловим потоком з таблиці “Параметри ламп розжарювання загального призначення з розрахунковими напругами 130 и 220 В” обираємо лампу Б215-225-75 потужністю 75 Вт, що має світловий потік 950 лм, найбільш близький до розрахункового.

Фактична освітленість:

$$E_f = E_{\min} * \Phi_{\text{л}} / \Phi_{\text{р}} = 75 * 950 / 741 = 96,15 \text{ лк.}$$

Загальна потужність:

$$P_{\text{общ}} = P_{\text{л}} * N = 75 * 4 = 300 \text{ Вт} = 0,3 \text{ кВт}.$$

Таким чином, розрахунок загального освітлення для офісного приміщення можна вважати виконаним, а його результати такими, що задовольняють загальноприйнятим нормам промислового освітлення.

5.4 Розробка заходів щодо зменшення впливу небезпечних і шкідливих факторів

Нормована освітленість робочого столу з ПК повинна бути в межах 300-500 лк. Допускається установка світильників місцевого освітлення для підсвічування документів. При цьому місцеве освітлення так само, як і загальне, не повинно створювати відблисків на екрані відеомоніторів. Зовнішня освітленість екрана монітора не повинна перевищувати 300 лк.

Бажано, щоб природне світло падало ліворуч, хоча допускається і таке розміщення робочого місця, коли віконні прорізи розташовані праворуч. Вікна повинні бути обладнані пристроями типу жалюзі, зовнішніх козирків і інше.

Зниження шуму, створюваного ПК, є дуже важливою задачею. Зниження шуму в джерелі випромінювання можна забезпечити застосуванням пружних прокладок між підставою ПК і опорною поверхнею. Як прокладки використовуються гума, пробка, різної конструкції амортизатори. Під настільні шумливі апарати можна підкладати м'які килимки із синтетичних матеріалів, а під ніжки столів, на яких вони встановлені, прокладки з м'якої гуми, повсті

товщиною 6 - 8 мм. Кріплення прокладок можливо шляхом приклейки їх до опорних частин.

Можливо також застосування звукоізолюючих кожухів, що не заважають технологічному процесові. Не менш важливим для зниження шуму в процесі експлуатації є питання правильного і своєчасного регулювання, змазування і заміни механічних вузлів шумливого устаткування.

Раціональне планування приміщення, розміщення устаткування є важливим чинником, що дозволяє знизити шум при існуючому устаткуванні ПК. Приміщення для сервісної апаратури необхідно розташовувати удалині від шумливого і вібруючого устаткування.

Зниження рівня шуму, що проникає у виробниче приміщення ззовні, може бути досягнуто збільшенням звукоізоляції конструкцій, що обгороджують, ущільненням по периметру притворів вікон, дверей.

У такий спосіб для зниження шуму створюваного на робочих місцях внутрішніми джерелами, а також шуму, що проникає ззовні, потрібно:

- послабити шум самих джерел (застосування екранів, звукоізолюючих кожухів);
- знизити ефект сумарного впливу відбитих звукових хвиль (звуковбирні поверхні конструкцій);
- застосовувати раціональне розташування устаткування;
- використовувати архітектурно-планувальні і технологічні рішення ізоляцій джерел шуму.

До робочого столу теж пред'являються певні вимоги. Його конструкція повинна забезпечувати оптимальне розміщення на його площі використовуваного устаткування. Робочий стілець користувача необхідно розташувати з таким розрахунком, щоб він дозволяв змінювати позу працюючого,

а також був підйомно-поворотним. Це допоможе уникнути стомлення м'язів шийно-плечової області і спини. Оптимальна відстань від очей до екрана монітора – 600-700 мм, але не ближче 500 мм з урахуванням розмірів алфавітно-цифрових знаків і символів. Недотримання цього параметра може призвести до швидкого розвитку стомлення і перевтоми зорового аналізатора і зі збільшенням тривалості роботи сприяти розвитку короткозорості. Відстань між робочими столами з відеотерміналами повинне бути не менш 2,0 м, а відстань між бічними поверхнями відеомоніторів – не менш 1,2 м. Робочі місця з ПК у залах електронно-обчислювальних машин чи у приміщеннях із джерелами шкідливих виробничих факторів розміщаються в ізольованих кабінах з організованим повітрообміном.

6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ

Охорона навколишнього природного середовища, раціональне використання природних ресурсів, забезпечення екологічної безпеки життєдіяльності людини – невід'ємна умова сталого економічного та соціального розвитку України.

З цією метою Україна здійснює на своїй території екологічну політику, спрямовану на збереження безпечного для існування живої і неживої природи навколишнього середовища, захисту життя і здоров'я населення від негативного впливу, зумовленого забрудненням навколишнього природного середовища, досягнення гармонійної взаємодії суспільства і природи, охорону, раціональне використання і відтворення природних ресурсів.

Відносини у галузі охорони навколишнього природного середовища в Україні регулюються Законом України № 1268-ХІІ від 26.06.91 р. «Про охорону навколишнього природного середовища», а також розроблюваними відповідно до нього земельним, водним, лісовим законодавством, законодавством про надра, про охорону атмосферного повітря, про охорону і використання рослинного і тваринного світу та іншим спеціальним законодавством.

Цей Закон визначає правові, економічні та соціальні основи організації охорони навколишнього природного середовища в інтересах нинішнього і майбутніх поколінь.

Завданням законодавства про охорону навколишнього природного середовища є регулювання відносин у галузі охорони, використання і відтворення природних ресурсів, забезпечення екологічної безпеки, запобігання і ліквідації негативного впливу господарської та іншої діяльності на навколишнє природне середовище, збереження природних ресурсів, генетичного фонду живої природи, ландшафтів та інших природних комплексів, унікальних територій та природних об'єктів, пов'язаних з історико-культурною спадщиною.

Охорона навколишнього природного середовища є системою державних і суспільних заходів, направлених на збереження, відтворювання і раціональне використання природних ресурсів і поліпшення полягання природного середовища, і є частиною прикладної екології.

В міру прискорення темпів науково-технічного прогресу дія людей на природу стає все більш могутньою. І в даний час воно вже сумирно із дією природних чинників, що приводить до якісної зміни співвідношення сил між суспільством і природою. На сучасному етапі людство поставлено перед чинником виникнення в природі незворотних процесів, нових шляхів переміщення і перетворення енергії і речовини. В природу втручається все більше і більше нових речовин, чужих їй, часом сильно токсичних для організмів. Частина з них не включається в природний круговорот і нагромаджується в біосфері, що приводить до небажаних екологічних наслідків.

Накопичення промислових відходів сприяє підвищенню захворюваності людей і тварин, прискоренню корозії машин і металевого устаткування, зниженню врожайності сільськогосподарських культур і продуктивності тваринництва, прискореному і нераціональному використуванню природних ресурсів і енергії, погіршенню багатьох властивостей екологічних систем,

загибелі деяких унікальних природних територіальних комплексів, зникненню окремих видів тварин і рослин.

Основну загрозу для навколишнього середовища в межах галузі розробки ПЗ становлять комплекси електронно-обчислювальних машин (надалі ЕОМ), а саме електромагнітні випромінювання від них.

Дослідження останніх років показали, що електромагнітні випромінювання вищезгаданих електронних пристроїв містять торсіонову компоненту, котра несе інформацію про процеси, що відбуваються в тому чи іншому електронному пристрої.

Торсіонові поля мають високу проникаючу здатність і їх неможливо заекранувати.

Останніми роками при проектуванні (конструюванні), виготовленні (будівництві) і експлуатації технічних і систем управління в різних сферах діяльності надзвичайно широко застосовуються персональні комп'ютери і всілякі комп'ютерні програми.

Робота з комп'ютерами програмістів, операторів і інших користувачів пов'язана з додатковими шкідливими і небезпечними чинниками, що негативно впливають на організм людини. Наприклад, шкідлива дія на зір надає не при тримання стандартних візуальних ергономічних параметрів екрану, розмір мінімального елементу відображення, мерехтіння зображення, відбивна здатність (відблиски) і ін. Працюючий комп'ютер створює електромагнітне поле, що шкідливо діє на організм людини. Це поле може викликати радіоперешкоди, тобто заважати роботі радіо і телеапаратури, що призводить до зниження надійності технічної системи або системи управління, до збільшення ризику виникнення аварійної ситуації у виробничому середовищі. Для забезпечення безпеки роботи з комп'ютером розроблені і повинні повсюдно

застосовуватися стандарти на монітори, вимоги до приміщень для експлуатації комп'ютерів і до організації і устаткування робочих місць.

6.2 Забруднення навколишнього середовища підприємством з розробки та супроводження програмного забезпечення

Хоча підприємством з розробки та супроводження програмного забезпечення не займається промисловим виробництвом з викидом шкідливих речовин у атмосферу, воду та ґрунт, це не виключає існування джерел забруднення довкілля. До таких джерел, насамперед, належать:

- дизельна електростанція;
- твердопаливні котли;
- побутове сміття;
- електромагнітне випромінювання.

Одночасно в офісі компанії можуть працювати до 100 комп'ютерів, стаціонарних телефонів та велика кількість серверної техніки, яка обслуговує роботу цього обладнання, аварійне чи планове відключення електроенергії призводить до значних перешкод у роботі персоналу. Для того, щоб уникнути таких ситуацій, на території компанії знаходиться дизельна електростанція. Вона працює як в аварійному, так і в резервному режимі.

Результатом роботи дизельної електростанції є продукти згоряння дизельного палива та шумове забруднення.

Твердопаливні котли використовуються в холодну пору року в системі опалення офісних приміщень. Для опалення використовуються деревні пелети. Хоча таке джерело тепла має менший відсоток шкідливих викидів в атмосферу,

ніж, наприклад, вугілля чи мазут, цей вид опалення не можна вважати екологічно чистим. Порівняльні характеристики продуктів згоряння палива наведені в табл. 6.1.

Таблиця 6.1

Рівні викидів забруднюючих речовин в атмосферу при спалюванні різних видів палива

Вид палива	Викиди забруднюючих речовин в атмосферне повітря без систем очищення, тонн на 1 тис. тонн нат. палива				
	CO ₂	NO ₂	SO ₂	Тверді частинки (пил неорг.)	РАЗОМ
Природний газ	1,18	3,52	0,00	0,00	4,70
Древні брикети, пелети	4,68	9,31	0,28	4,11	17,69
Деревина дров'яна	4,9	9,4	0,3	4,3	18,9
Тирса деревна	5,0	9,6	0,5	5,0	20,0
Древні відходи, обрізки	5,2	9,9	0,4	5,2	20,7
Швидкозростаюча деревина	4,8	9,5	0,0	8,4	22,7
Тріска, сучки, кора	5,6	11,4	0,8	13,4	31,3
Мазут	5,20	5,20	35,30	0,30	45,90
Брикет торф'яний	8,04	26,81	3,00	13,02	50,87
Кам'яне вугілля	9,58	63,56	9,20	65,32	147,66

Побутові відходи – це залишки речовин і предметів, які утворюються в результаті побутової та господарської діяльності людини і які не можуть бути використані на місці утворення, а їх накопичення і зберігання порушують санітарний стан навколишнього середовища.

Всі побутові відходи, які утворюються в процесі функціонування компанії, можна розділити на рідкі та тверді. До рідких побутових відходів належать нечистоти з вигребів туалетів, помий (від миття посуду, підлоги, тощо) і стічні води (побутові, зливові). До твердих побутових відходів можна віднести сміття (побутові відходи) та кухонні відходи. Зокрема, твердими відходами є папір, картон, скло, пластмаса, продукти харчування.

Генератором електромагнітного забруднення є, в першу чергу, велика кількість комп'ютерної техніки на території офісу. Більшість із них працює 8 годин на добу, а деякі не вимикаються цілодобово. Випромінювачами в даному випадку є і процесор, і монітор. Випромінювання останнього значно вищі, особливо його бічні і задні стінки, адже вони не мають спеціального захисного покриття, як у лицьовій частині монітора. Крім того, в офісі є дві кухні, кожна з яких обладнана 5 мікрохвильовими печами.

Електромагнітне випромінювання має несприятливий вплив на організм людини. Його наслідками можуть бути головний біль, порушення сну, перевтома і, навіть, у деяких випадках стенокардія.

6.3 Розробка заходів щодо зменшення забруднення

Для того, щоб зменшити рівень забруднення навколишнього середовища, варто вжити певних заходів, які мінімізують цей вплив, якщо його неможливо усунути цілком. Серед таких заходів:

- 1) Очищення димових газів у мокрих золоуловлювачах.

Електрофільтри на електростанціях застосовуються для досягнення найбільш глибокого очищення димових газів в основному на великих

енергоблоках потужністю 300 МВт та більше. Мокрі золоуловлювачі, які працюють при маленьких питомих витратах води та невеликих перепадах тиску, встановлюються за котлоагрегатами середньої паропроductивності. У мокрих золоуловлювачах уловлювання часток золи на плівці води, яка стікає по його стінках, здійснюється за рахунок відцентрової сили, яка діє на частки. Ефективність апарата не перевищує 90 %.

2) Використання природного газу для опалення офісних приміщень замість деревних пелетів.

Згідно з показниками, наведеними в табл. 6.1, рівень шкідливих викидів від згоряння природного газу є найнижчим серед перерахованих видів палива. Проте зміну джерела енергії слід проводити, враховуючи економічну ефективність, оскільки вартість цих видів палива не є однаковою.

3) Раціональне позбавлення від побутових відходів.

Деякі побутові відходи можна безпечно спалювати для отримання енергії. Вторинну сировину (макулатуру, скло, пластмаси) можна відправляти на переробку, а не вивозити на сміттєзвалища. Крім того, зменшити кількість побутових відходів допоможе повторне їх використання (наприклад, скляних чи пластикових пляшок та контейнерів для їжі), скорочення споживання та використання меншої кількості упаковки.

4) Зменшити рівень електромагнітного забруднення та захиститися від його надмірного впливу.

Зрозуміло, що неможливо повністю обійтися без електроприладів та комп'ютерів, проте можна дещо зменшити їх негативний вплив. Для цього слід вимикати з електромережі комп'ютери, телефони та обладнання, з якими ніхто не працює, виходити на прогулянки і робити перерви в роботі з комп'ютером.

ВИСНОВКИ

В кваліфікаційній роботі вирішене наукове завдання з підвищення достовірності оцінювання тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК за допомогою нелінійної регресійної моделі.

1) Здійснено аналіз існуючих регресійних рівняння для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК. З'ясовано, на сьогоднішній день існують нелінійні регресійні рівняння для оцінювання тривалості розробки застосунків, що створюються для ПК. Як показав аналіз даних репозитарію ISBSG з релізу 2021 року, поділ моделей тільки за комп'ютерними платформами вже є недостатнім, оскільки отримані моделі мають погані значення показників якості, таких як R^2 , PRED(0,25) та MMRE. Тому окрім комп'ютерної платформи виникає необхідність додавання інших ознак кластеризації даних. Однією з таких ознак окрім комп'ютерної платформи природньо розглядати ще і мову програмування. Це призводить до необхідності використання нелінійних регресійних рівнянь для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

2) Здійснено підбір сукупності даних, за якими будується нелінійна створюються мовою Visual Basic для ПК, та перевірку цих даних на наявність двовимірних викидів. З'ясовано, застосування критерія на основі квадрата Махаланобіса до нормалізованих даних показало, що дані не містять двовимірних викидів.

3) Удосконалено модель ISBSG для оцінювання тривалості розробки ПЗ для ПК за рахунок, по-перше, врахування додаткової ознаки – мови програмування Visual Basic, а, по-друге, використання замість нелінійного регресійного рівняння нелінійної регресійної моделі, яка у своєму складі містить дві випадкові величини: тривалість та похибку. Ця нелінійна регресійна модель в порівнянні з моделлю ISBSG, яка побудована на основі нормалізуючого перетворення у вигляді десяткового логарифму без врахування викидів у регресії, має більші значення множинного коефіцієнту детермінації і відсотка прогнозування, менші значення середньої величини відносної похибки та ширин довірчого інтервалу і інтервалу прогнозування для більшої кількості значень незалежних змінних.

4) На підставі запропонованої нелінійної регресійної моделі розроблено програму для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК для системи моделювання Scilab 6.0.0. Була розроблена наступна програмна документація: технічне завдання, текст програми, опис програми, інструкція користувача, програма та методика випробувань, що наведені відповідно у додатках А, Б, В, Г, Д. Були проведені випробування програми згідно з програмою та методикою випробувань, що наведена у додатку Д. За результатами випробувань зроблено висновок про працездатність розробленої програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК, для системи моделювання Scilab 6.0.0.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Trendowicz A. Software project effort estimation. foundations and best practice guidelines for success / A. Trendowicz, R. Jeffery. – Springer International Publishing, 2014. – 491 p. DOI: <https://doi.org/10.1007/978-3-319-03629-8>.
2. Boehm B.W. Software engineering economics [Text] / B. W. Boehm. – Englewood Cliffs, NJ : Prentice Hall, 1981. – 768 p. – ISBN 0-13-822122-7.
3. Oligny S., Bourque P., Abran A. & Fournier B. (2000). Exploring the relation between effort and duration in software engineering projects. *Proc. of the 16th IFIP World Computer Congress* (pp. 175-178). Aug, 2000, Beijing, China
4. Приходько С.Б. Розробка нелінійної регресійної моделі тривалості програмних проєктів на основі нормалізуючого перетворення Джонсона / С. Б. Приходько, А. В. Пухалевич // Радіоелектронні і комп'ютерні системи. – Харків, 2012. – № 4 (56) – С.90-93. – ISSN: 1814-4225
5. C. López-Martín, A. Chavoya and M. E. Meda-Campaña, "Use of a Feedforward Neural Network for Predicting the Development Duration of Software Projects," *2013 12th International Conference on Machine Learning and Applications*, 2013, pp. 156-159, doi: 10.1109/ICMLA.2013.182.
6. Prykhodko S. Developing PC Software Project Duration Model based on Johnson transformation / Sergiy Prykhodko, Andriy Pukhalevich // 12th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science TCSET'2014 25 February – 1 March 2014, Lviv-Slavske, Ukraine. – Lviv Polytechnic National University, 2014. – P.114-116. – ISBN: 978-617-607-208-9
7. Prykhodko S.B. Confidence interval estimation of PC software project duration regression based on Johnson transformation / S. B. Prykhodko,

A. V. Pukhalevich // Радіоелектронні і комп'ютерні системи. – Харків, 2014. – № 2 (66) – С.104-107. – ISSN: 1814-4225

8. Приходько С.Б. Розробка нелінійних регресійних моделей тривалості програмних проєктів на основі перетворення Джонсона / С. Б. Приходько, А. В. Пухалевич // Зб. наук. праць НУК. – Миколаїв: НУК, 2014. – № 3. – С.76-80. DOI: <http://dx.doi.org/10.15589/jnn20140312>

9. C. López-Martín, "Feedforward Neural Networks for Predicting the Duration of Maintained Software Projects," *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2016, pp. 528-533, DOI: <http://dx.doi.org/10.1109/ICMLA.2016.0093>.

10. Приходько С.Б. Інформаційна технологія для оцінювання тривалості проєктів з розробки програмного забезпечення / С. Б. Приходько, А. В. Пухалевич // Проблемы информационных технологий. – 2016. – № 1 (019). – С.81-87. – ISSN 1998-7005

11. C. Lopez-Martin, S. Banitaan, A. Garcia-Floriano and C. Yanez-Marquez, "Support Vector Regression for Predicting the Enhancement Duration of Software Projects," *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2017, pp. 562-567, DOI: <http://dx.doi.org/10.1109/ICMLA.2017.0-101>.

12. Приходько С.Б. Нелінійні регресійні моделі для оцінювання тривалості розробки програмних застосунків, що створюються мовами Java та Visual Basic для ПК / С. Б. Приходько, А. В. Пухалевич, **І. П. Халанчук** // Інформаційні технології: моделі, алгоритми, системи (ITMAS – 2021): Матеріали II-ої Всеукраїнської науково-практичної інтернет конференції (26-28 жовтня 2021 р.). – Миколаїв: НУК імені адмірала Макарова, 2021. – С. 6-7. Режим доступу: <http://itconf.nuos.edu.ua/2021/proceedings/>

13. TIOBE Index for November 2021. November Headline: PHP about to lose its top 10 position. URL: <https://www.tiobe.com/tiobe-index/> (дата звернення: 01.12.2021)

14. Приходько С.Б. Порівняння ймовірнісних моделей тривалості робіт в програмних проектах / С. Б. Приходько, А. В. Пухалевич // Восточно-Европейский журнал передовых технологий // научный журнал. – Харьков: Технологический центр, 2012. – №1/13 (55). – С.39-41. – ISSN: 1729-3774

15. Prykhodko N.V. Constructing the non-linear regression models on the basis of multivariate normalizing transformations / N. V. Prykhodko, S. B. Prykhodko // Electronic modeling. – 2018. – Т.40. – № 6. – Р. 99-108. – ISSN 0204–3572. DOI: <https://doi.org/10.15407/emodel.40.06.101>

16. Приходько С.Б. Метод покращення нелінійних регресійних моделей на основі багатовимірних нормалізуючих перетворень / С. Б. Приходько, Н. В. Приходько // Прикладні науково-технічні дослідження: матеріали III міжнар. наук.-практ. конф., 3-5 квіт. 2019 р. – Академія технічних наук України. – Івано-Франківськ: Симфонія форте, 2019. – С. 20. – ISBN 978-966-284-132-9

17. Prykhodko S. Mathematical Modeling of Non-Gaussian Dependent Random Variables by Nonlinear Regression Models Based on the Multivariate Normalizing Transformations / S. Prykhodko, N. Prykhodko // In: Shkarlet S., Morozov A., Palagin A. (eds) Mathematical Modeling and Simulation of Systems (MODS'2020). MODS 2020. Advances in Intelligent Systems and Computing, vol. 1265, 2021. – P. 166-174. Springer, Cham. DOI: https://doi.org/10.1007/978-3-030-58124-4_16

18. Prykhodko S. Detecting Outliers in Multivariate Non-Gaussian Data on the basis of Normalizing Transformations / S. Prykhodko, N. Prykhodko, L. Makarova, K. Pugachenko // Proceedings of the 2017 IEEE First Ukraine Conference on

Electrical and Computer Engineering (UKRCON) «Celebrating 25 Years of IEEE Ukraine Section», Kyiv, Ukraine, May 29 – June 2, 2017. – P. 846-849.

19. Prykhodko S. Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate Non-Gaussian Data / S. Prykhodko, N. Prykhodko, L. Makarova, A. Pukhalevych // Proceedings of 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, February 20–24, 2018. – P. 962-965.

20. Johnson R.A. Applied Multivariate Statistical Analysis / R. A. Johnson, D. W. Wichern. – Pearson Prentice Hall, 2007. – 800 p.

21. Foss T. A simulation study of the model evaluation criterion MMRE / T. Foss, E. Stensrud, B. Kitchenham, I. Myrtveit // IEEE Transactions on software engineering. – 2003. – 11(29). – P. 985–995.

22. Port, D. Comparative studies of the model evaluation criterions MMRE and PRED in software cost estimation research / D. Port, M. Korte // In: Proceedings of the 2nd ACM-IEEE International Symposium on Empirical Software Engineering and Measurement. – ACM, New York, 2008. – P. 51–60.

23. Campbell S.L. Modeling and Simulation in Scilab/Scicos [Text] / Stephen L. Campbell, Jean-Philippe Chancelier, Ramine Nikoukhah. – Springer, 2005. – 313 p.

24. Алексеев Е.Р. Scilab: Решение инженерных и математических задач [Текст] / Е. Р. Алексеев, О. В. Чеснокова, Е. А. Рудченко. – М.: ALT Linux; БИНОМ. Лаборатория знаний, 2008. – 269 с.

Додаток А – Технічне завдання на розробку програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Вступ

Назва розробки – “Програма для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК”, далі програма.

Підстави для розробки

Завдання на дипломну роботу.

Призначення розробки:

Програма призначена для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК за нелінійним рівнянням регресії.

Вимоги до програми:

Вимоги функціональних характеристик.

Вимоги до складу функцій, що виконуються.

Програма має виконувати такі функції:

1) Вводити з текстового файлу емпіричні дані з метрик програмних застосунків, що створені мовою Visual Basic для ПК (значення випадкових величин): тривалість розробки застосунків у місяцях Y та трудомісткість розробки у людино-годинах X_1 .

2) Вводити з інтерфейсу програми довірчу ймовірність, трудомісткість (у людино-годинах) розробки програмних застосунків, що створюються мовою Visual Basic для ПК, за якими потрібно оцінити тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.

3) Виконувати нормалізацію емпіричних даних за нормалізуючим перетворенням у вигляді десяткового логарифму.

4) Знаходити значення коефіцієнтів лінійної регресійної моделі за нормалізованими емпіричними даними.

5) Здійснювати побудову нелінійної регресійної моделі.

6) Обчислювати значення коефіцієнту детермінації R^2 .

7) Здійснювати побудову інтервалу прогнозування нелінійної регресії.

8) Виводити на екран нелінійну регресію, довірчий інтервал та інтервал прогнозування нелінійної регресії тривалості (у місяцях) розробки програмних застосунків, що створюються мовою Visual Basic для ПК, значення коефіцієнту детермінації R^2 .

9) Виводити у текстовий файл результати оцінювання тривалості (у місяцях) розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

Вимоги до організації вхідних та вихідних даних

Вхідні дані:

- емпіричні дані з метрик програмних застосунків, що створені мовою Visual Basic для ПК (значення випадкових величин): перша метрика Y включає тривалість розробки застосунків у місяцях та друга метрика X_1 визначає трудомісткість розробки у людино-годинах;

- значення довірчої ймовірності та трудомісткість розробки у людино-годинах, за якою потрібно оцінити тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Visual Basic для ПК.

Вихідні дані:

- значення нелінійної регресії та границь довірчого інтервалу і інтервалу прогнозування нелінійної регресії тривалості розробки програмних застосунків, що створюються мовою Visual Basic для ПК.

- значення коефіцієнту детермінації.

Вимоги до надійності не висуваються.

Вимоги до умов експлуатації не висуваються.

Вимоги до складу й параметрів технічних засобів.

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2, оперативна пам'ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Вимоги до інформаційної та програмної сумісності.

Для запуску та роботи з програмним забезпеченням необхідна операційна система Windows Vista/7/8/10 (32 або 64 біти), система моделювання Scilab 6.0.0., компілятор C (Visual Studio 2010 або Visual Express 2010) для компіляції зовнішніх модулів C або C++ та використання моделі в Xcos.

Вимоги до маркування та пакування.

Визначаються вимогами до пакування CD дисків.

Вимоги до транспортування та зберігання.

Програма може зберігатись та транспортуватися на CD дисках.

Вимоги до програмної документації.

Програмна документація має включати: технічне завдання на розробку програми, текст програми, опис програми, інструкція користувача, програма та методика випробувань.

Техніко-економічні показники не розраховуються.

Стадії та етапи розробки.

Стадії та етапи розробки програми представлені у таблиці.

Таблиця – Стадії та етапи розробки

Стадія розробки	Етапи робіт	Термін виконання
1. Технічне завдання	Розробка та затвердження технічного завдання	21.09.2021
2. Ескізний проект	Розробка ескізного проекту	24.09.2021
	Затвердження ескізного проекту	26.09.2021
3. Технічний проект	Розробка технічного проекту	08.10.2021
	Затвердження технічного проекту	10.10.2021
4. Робочий проект	Розробка програми	05.11.2021
	Розробка програмної документації	12.11.2021
	Випробування програми	21.11.2021

Порядок контролю і приймання

Порядок контролю та приймання програми здійснюється представниками замовника у присутності представника розробника.

Для приймання програми надається програмна документація, яка була розроблена відповідно до технічного завдання. Представники замовника установлюють відповідність програми технічному завданню.

За результатами приймання програми представниками замовника складається акт приймання програми.

Додаток Б – Текст програми

```
//Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic
для ПК
f=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f,'position',[0,0,700,300])
//Устанавливаем заголовок окна.
set(f,'figure_name',...
'Oцінювання тривалості розробки застосунків, що створюються мовою Visual Basic
для ПК');

//plot2d(x,y,-9);

lab_text0=uicontrol(f,'style','text','string',...
'Файл з даними','position',...
[350,270,80,20]);
edit_text=uicontrol(f,'style','edit','string',...
'D:\Data_F\VB_Duraton_39_2.txt','position',[320,240,140,20]);

BtDatInp=uicontrol('style','pushbutton','string','Ввести',...
'CallBack','[x,yx,f1,f2,f3,n]=DataInput()','position',[340,210,100,20]);

//lab_text1=uicontrol(f,'style','text','string',...
//'Параметри нормалізуючого перетворення Джонсона','position',...
//[100,270,300,20]);
lab_text2=uicontrol(f,'style','text','string',...
'Наявність фактичної тривалості у місяцях','position',...
[475,270,215,20]);
lab_text3=uicontrol(f,'style','text','string',...
'Дані для оцінювання','position',...
[60,270,110,20]);
lab_text4=uicontrol(f,'style','text','string',...
'Трудомісткість розробки (людино-год.)','position',...
[20,240,200,20]);
edit_text4=uicontrol(f,'style','edit','string',...
'512','position',[230,240,60,20]);
//lab_text5=uicontrol(f,'style','text','string',...
//'Загальна кількість співвідношень','position',...
//[20,210,200,20]);
//edit_text5=uicontrol(f,'style','edit','string',...
//'2','position',[230,210,60,20]);
//lab_text6=uicontrol(f,'style','text','string',...
//'Середня кількість атрибутів на клас','position',...
//[20,180,200,20]);
//edit_text6=uicontrol(f,'style','edit','string',...
//'10.6','position',[230,180,60,20]);
lab_text7=uicontrol(f,'style','text','string',...
'Довірча ймовірність','position',...
[20,210,200,20]);
//[20,150,200,20]);
edit_text7=uicontrol(f,'style','edit','string',...
'0.95','position',[230,210,60,20]);
```

```

// '0.95', 'position', [230, 150, 60, 20]);
lab_text8=uicontrol(f, 'style', 'text', 'string', ...
'Reзультати оцінювання', 'position', ...
[50, 120, 120, 20]);
lab_text9=uicontrol(f, 'style', 'text', 'string', ...
'Reзультат нел. регресії тривалості (місяці)', 'position', ...
[20, 90, 245, 20]);
lab_text10=uicontrol(f, 'style', 'text', 'string', ...
'Нижня границя довір. інтервалу тривалості (міс.)', 'position', ...
[20, 60, 245, 20]);
lab_text11=uicontrol(f, 'style', 'text', 'string', ...
'Верхня границя довір. інтервалу тривалості (міс.)', 'position', ...
[20, 30, 245, 20]);
lab_text12=uicontrol(f, 'style', 'text', 'string', ...
'Нижня границя інтервалу прогноз. тривалості (міс.)', 'position', ...
[355, 60, 260, 20]);
lab_text13=uicontrol(f, 'style', 'text', 'string', ...
'Верхня границя інтервалу прогноз. тривалості (міс.)', 'position', ...
[355, 30, 260, 20]);
//
//Текстовое поле, определяющее вывод результатов.
textresult=uicontrol(f, 'style', 'text', 'string', '', 'position', ...
[270, 90, 70, 20]);
textresult2=uicontrol(f, 'style', 'text', 'string', '', 'position', ...
[270, 60, 70, 20]);
textresult3=uicontrol(f, 'style', 'text', 'string', '', 'position', ...
[270, 30, 70, 20]);
textresult4=uicontrol(f, 'style', 'text', 'string', '', 'position', ...
[190, 120, 250, 20]);
textresult5=uicontrol(f, 'style', 'text', 'string', '', 'position', ...
[620, 60, 70, 20]);
textresult6=uicontrol(f, 'style', 'text', 'string', '', 'position', ...
[620, 30, 70, 20]);

//Флажок, отвечающий за выбор типа преобразования.
radio_bikv_x=uicontrol('style', 'radiobutton', 'string', ...
'є (включити), немає (виключити)', 'value', 1, 'position', ...
[490, 240, 200, 20]);
//
//radio_bikv_y=uicontrol('style', 'radiobutton', 'string', ...
// 'SU (включити), SB (виключити)', 'value', 1, 'position', ...
//[500, 180, 300, 20]);
//
//Проверяем значение флажка, если флажок выключен,
//if (get(radio_bikv_x, 'value')==0) & (get(radio_bikv_y, 'value')==0)
if (get(radio_bikv_x, 'value')==0)

else
//   set(textresult, 'string', ...
//   'Нелінійне рівняння регресії не знайдено, Моделювання не виконано');
edit_text12=uicontrol(f, 'style', 'edit', 'string', ...
'0', 'position', [500, 210, 60, 20]);
end;

```

```

BtNr_pred_int=uicontrol('style','pushbutton','string','Оцінювання','CallBack',...
.
'VB_AppDurIntEst(x,yx,n)', 'position',[340,150,100,20]);

BtClose=uicontrol('style','pushbutton','string','Закрити',...
'CallBack','W_Close()','position',[350,5,100,20]);

function [x, yx, f1, f2, f3, n]=DataInput()
dir_name=get(edit_text,'string');

file_PC=mopen(dir_name,'r');
file_PC2=mopen('D:\Data_F\VB_Duraton_39_data2.txt','w+');
file_PC3=mopen('D:\Data_F\VB_Duraton_39_data3.txt','w+');
i=1;
//n=52;
while (feof(file_PC))<1
    y(i)=mfscanf(file_PC,'%f'); x1(i)=mfscanf(file_PC,'%f');
    //x2(i)=mfscanf(file_PC,'%f'); x3(i)=mfscanf(file_PC,'%f');
    i=i+1;
end;
n=i-1;

fclose(file_PC);

for i = 1:n
    yx(i,1)=y(i);
    //yx(i,2)=x1(i); yx(i,3)=x2(i); yx(i,2)=x3(i);
    x(i,1)=x1(i);
    //x(i,2)=x2(i); x(i,3)=x3(i);
end;
//for i=1:n
    fprintf(file_PC2,'%f\t%f\n',y,x1);
    //fprintf(file_PC2,'%f\t%f\t%f\t%f\n',y,x1,x2,x3);
//end;

for i = 1:n
    for j = 1:1 fprintf(file_PC3, "%5.3f ", x(i,j)); end;
    fprintf(file_PC3, "\n");
end;

fclose(file_PC2);
fclose(file_PC3);
f1=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f1,'position',[50,50,700,300]);
//Устанавливаем заголовок окна.
set(f1,'figure_name','Емпіричні дані: тривалість (у місяцях) в залежності від
трудомісткості (у людино-годинах)');
plot2d(x1,y,-9);

f2=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f2,'position',[50,50,700,300]);
//Устанавливаем заголовок окна.

```

```

set(f2,'figure_name','Емпіричні дані: розмір (у тисячах строк коду) в залежності
від загальної кількості співвідношень');
//plot2d(x2,y,-9);

f3=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f3,'position',[50,50,700,300]);
//Устанавливаем заголовок окна.
set(f3,'figure_name','Емпіричні дані: розмір (у тисячах строк коду) в залежності
від середньої кількості атрибутів на клас');
//plot2d(x3,y,-9);

x1A=0; x2A=0; x3A=0;
for i=1:n
    x1A=x1A+x(i,1);
    //x2A=x2A+x(i,2); x3A=x3A+x(i,3);
end;
x1A=x1A/n; x2A=x2A/n; x3A=x3A/n;
Sx1x1=0; Sx2x2=0; Sx3x3=0;
Sx1x2=0; Sx1x3=0; Sx2x3=0;
for i=1:n
    Sx1x1=Sx1x1+(x(i,1)-x1A)*(x(i,1)-x1A);
    //Sx1x2=Sx1x2+(x(i,1)-x1A)*(x(i,2)-x2A);
    //Sx1x3=Sx1x3+(x(i,1)-x1A)*(x(i,3)-x3A);
    //Sx2x3=Sx2x3+(x(i,2)-x2A)*(x(i,3)-x3A);
    //Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    //Sx3x3=Sx3x3+(x(i,3)-x3A)*(x(i,3)-x3A);
end;
Sxx(1,1)=Sx1x1;
//Sxx(1,2)=Sx1x2; Sxx(1,3)=Sx1x3;
//Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2; Sxx(2,3)=Sx2x3;
//Sxx(3,1)=Sx1x3; Sxx(3,2)=Sx2x3; Sxx(3,3)=Sx3x3;

//Sxx_1=1/Sxx;

z1A=0; z2A=0; z3A=0; zYA=0;
for i=1:n
    z1A=z1A+log10(x(i,1));
    //z2A=z2A+log10(x(i,2)); z3A=z3A+log10(x(i,3));
    zYA=zYA+log10(yx(i,1));
end;
z1A=z1A/n;
//z2A=z2A/n; z3A=z3A/n;
zYA=zYA/n;
Sz1z1=0; Sz2z2=0; Sz3z3=0;
Sz1z2=0; Sz1z3=0; Sz2z3=0;
SzYz1=0;
for i=1:n
    SzYz1=SzYz1+(log10(yx(i,1))-zYA)*(log10(x(i,1))-z1A);
    Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    //Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    //Sz1z3=Sz1z3+(log10(x(i,1))-z1A)*(log10(x(i,3))-z3A);
    //Sz2z3=Sz2z3+(log10(x(i,2))-z2A)*(log10(x(i,3))-z3A);
    //Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);

```

```

        //Sz3z3=Sz3z3+(log10(x(i,3))-z3A)*(log10(x(i,3))-z3A);
end;
//Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2; Szz(1,3)=Sz1z3;
//Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2; Szz(2,3)=Sz2z3;
//Szz(3,1)=Sz1z3; Szz(3,2)=Sz2z3; Szz(3,3)=Sz3z3;

//Szz_1=1/Szz;

//zi(1)=log10(x(1,1))-z1A; zi(2)=log10(x(1,2))-z2A; zi(3)=log10(x(1,3))-z3A;
//ziT=zi';
//ziSzz_1=ziT*Szz_1;
//result=ziSzz_1*zi;
zi_res=log10(x(1,1))-z1A;
result=zi_res*zi_res/Sz1z1;

Df=n-2;
P=0.975;
Q=1-P;

[T]=cdf('T',Df,P,Q);

//      Delta_z=T/n^0.5;

b1=SzYz1/Sz1z1;
b0=zYA-b1*z1A;

Szyzyn4=0;
for i=1:n
    Szyzyn4=Szyzyn4+...
        (b0+b1*log10(x(i,1))-log10(yx(i,1)))^2;
    //(b0+b1*log10(x(i,1))+b2*log10(x(i,2))+b3*log10(x(i,3))-log10(yx(i,1)))^2;
end;
Szyzyn4=Szyzyn4/(n-2)
Delta_z=T*(Szyzyn4*(1/n+result))^0.5;

//zy_result=b0+b1*log10(x(1,1))+b2*log10(x(1,2))+b3*log10(x(1,3));
zy_result=b0+b1*log10(x(1,1))

zyLBC=zy_result-Delta_z; zyUBC=zy_result+Delta_z;

y_result=10^zy_result;
yLBC=10^zyLBC; yUBC=10^zyUBC;

file_PC5=fopen('D:\Data_F\VB_Duration_39_data5.txt','w');
//for i=1:3
    fprintf(file_PC5, '%5.6f ', yUBC);
//end
fclose(file_PC5);

endfunction

//Функція оцінювання тривалості розробки застосунків,
//що створюються мовою Visual Basic для ПК
function VB_AppDurIntEst(x, yx, n)

```

```

//Считываем значение переменных из текстовых полей и
//преобразовываем их к числовому типу.
x1i=eval(get(edit_text4,'string'));
//x2i=eval(get(edit_text5,'string'));
//x3i=eval(get(edit_text6,'string'));
Pconf=eval(get(edit_text7,'string'));

x1A=0; x2A=0; x3A=0; yA=0;
for i=1:n
    x1A=x1A+x(i,1);
    //x2A=x2A+x(i,2); x3A=x3A+x(i,3);
    yA=yA+yx(i,1);
end;
x1A=x1A/n;
//x2A=x2A/n; x3A=x3A/n;
yA=yA/n;
Sx1x1=0; Sx2x2=0; Sx3x3=0;
Sx1x2=0; Sx1x3=0; Sx2x3=0; SyA=0;
for i=1:n
    Sx1x1=Sx1x1+(x(i,1)-x1A)*(x(i,1)-x1A);
    //Sx1x2=Sx1x2+(x(i,1)-x1A)*(x(i,2)-x2A);
    //Sx1x3=Sx1x3+(x(i,1)-x1A)*(x(i,3)-x3A);
    //Sx2x3=Sx2x3+(x(i,2)-x2A)*(x(i,3)-x3A);
    //Sx2x2=Sx2x2+(x(i,2)-x2A)*(x(i,2)-x2A);
    //Sx3x3=Sx3x3+(x(i,3)-x3A)*(x(i,3)-x3A);
    SyA=SyA+(yx(i,1)-yA)*(yx(i,1)-yA);
end;
//Sxx(1,1)=Sx1x1; Sxx(1,2)=Sx1x2; Sxx(1,3)=Sx1x3;
//Sxx(2,1)=Sx1x2; Sxx(2,2)=Sx2x2; Sxx(2,3)=Sx2x3;
//Sxx(3,1)=Sx1x3; Sxx(3,2)=Sx2x3; Sxx(3,3)=Sx3x3;

//Sxx_1=1/Sxx;

z1A=0; z2A=0; z3A=0; zYA=0;
for i=1:n
    z1A=z1A+log10(x(i,1));
    //z2A=z2A+log10(x(i,2)); z3A=z3A+log10(x(i,3));
    zYA=zYA+log10(yx(i,1));
end;
z1A=z1A/n;
//z2A=z2A/n; z3A=z3A/n;
zYA=zYA/n;
Sz1z1=0; Sz2z2=0; Sz3z3=0;
Sz1z2=0; Sz1z3=0; Sz2z3=0;
SzYz1=0;
for i=1:n
    SzYz1=SzYz1+(log10(yx(i,1))-zYA)*(log10(x(i,1))-z1A);
    Sz1z1=Sz1z1+(log10(x(i,1))-z1A)*(log10(x(i,1))-z1A);
    //Sz1z2=Sz1z2+(log10(x(i,1))-z1A)*(log10(x(i,2))-z2A);
    //Sz1z3=Sz1z3+(log10(x(i,1))-z1A)*(log10(x(i,3))-z3A);
    //Sz2z3=Sz2z3+(log10(x(i,2))-z2A)*(log10(x(i,3))-z3A);
    //Sz2z2=Sz2z2+(log10(x(i,2))-z2A)*(log10(x(i,2))-z2A);
    //Sz3z3=Sz3z3+(log10(x(i,3))-z3A)*(log10(x(i,3))-z3A);
end;

```

```

//Szz(1,1)=Sz1z1; Szz(1,2)=Sz1z2; Szz(1,3)=Sz1z3;
//Szz(2,1)=Sz1z2; Szz(2,2)=Sz2z2; Szz(2,3)=Sz2z3;
//Szz(3,1)=Sz1z3; Szz(3,2)=Sz2z3; Szz(3,3)=Sz3z3;

//Szz_1=1/Szz;

//zi(1)=log10(x(1,1))-z1A; zi(2)=log10(x(1,2))-z2A; zi(3)=log10(x(1,3))-z3A;
//ziT=zi';
//ziSzz_1=ziT*Szz_1;
//result=ziSzz_1*zi;
zi_res=log10(x(1,1))-z1A;
result=zi_res*zi_res/Sz1z1;

Df=n-2;
P=0.975;
Q=1-P;

[T]=cdf('T',Df,P,Q);

b1=SzYz1/Sz1z1;
b0=zYA-b1*z1A;

Szyzyn4=0; Syy=0;
for i=1:n
    Szyzyn4=Szyzyn4+...
        (b0+b1*log10(x(i,1))-log10(yx(i,1)))^2;

    Syy=Syy+(yx(i,1)-10^b0*x(i,1)^b1)^2;
end;
Szyzyn4=Szyzyn4/(n-2)
Delta_z=T*(Szyzyn4*(1/n+result))^0.5;
Delta_z_pr=T*(Szyzyn4*(1+1/n+result))^0.5;

r_2=1-Syy/SyA;

zy_result=b0+b1*log10(xli);

zyLBC=zy_result-Delta_z; zyUBC=zy_result+Delta_z;

zyLBPr=zy_result-Delta_z_pr; zyUBPr=zy_result+Delta_z_pr;

y_result=10^zy_result;
yLBC=10^zyLBC; yUBC=10^zyUBC;
yLBPr=10^zyLBPr; yUBPr=10^zyUBPr;

file_PC5=fopen('D:\Data_F\VB_Duration_39_data5.txt','w+');
//for i=1:3
    fprintf(file_PC5, "%5.6f ", yUBC);
//end
fclose(file_PC5);

f2=figure(); //Создание графического окна.
//Устанавливаем размеры окна.
set(f2,'position',[50,50,700,300]);

```

```

//Устанавливаем заголовок окна.
set(f2,'figure_name','Регресія тривалості (у місяцях) та її інтервали
прогнозування в залежності від трудомісткості (у людино-годинах)');

for i=1:n
    y(i)=y $x$ (i,1);
    x1(i)= $x$ (i,1);
    zr(i)=b0+b1*log10( $x$ (i,1));
    yr(i)=10^zr(i);
    zi_res=log10( $x$ (1,1))-z1A;
    result=zi_res*zi_res/Sz1z1;
    Delta_z=T*(Szyzyn4*(1+1/n+result))^0.5;

    pizrmin=zr(i)-Delta_z;
    pizrmax=zr(i)+Delta_z;
    piyr_min(i)=10^pizrmin;
    piyr_max(i)=10^pizrmax;

end;

plot2d(x1,piyr_max,5);
plot2d(x1,y,-9);
plot2d(x1,yr,1);
plot2d(x1,piyr_min,5);

set(textresult,'string', sprintf("=%1.4f",y_result));
set(textresult2,'string', sprintf("=%1.4f",yLBC));
set(textresult3,'string', sprintf("=%1.4f",yUBC));
set(textresult4,'string',...
    sprintf("Коефіцієнт детермінації R^2=%1.4f",r_2));
set(textresult5,'string', sprintf("=%1.4f",yLBPr));
set(textresult6,'string', sprintf("=%1.4f",yUBPr));

if (get(radio_bikv_x,'value')==0)
else
    yfact=eval(get(edit_text12,'string'));
    MRE=abs((yfact-y_result)/yfact);
    textresult5=uicontrol(f,'style','text','string','','position',...
        [500,120,160,20]);
    set(textresult5,'string',...
        sprintf("Відносна похибка MRE=%1.4f",MRE));
end;

endfunction

//Функция закрытия окна.
function W_Close()
close(f);
close(f1);
close(f2);
close(f3);

endfunction

```

ДОДАТОК В – Опис програми

Загальні відомості

Програма для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК для системи моделювання Scilab 6.0.0. Файл програми VB_AppDurEst.sce має розмір 15 КБ.

Функціональне призначення

Програма VB_AppDurEst призначений для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК для системи моделювання Scilab 6.0.0.

Технічні засоби, що використовуються

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2, оперативна пам'ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Виклик та завантаження

Для запуску та роботи з програмою необхідна операційна система Windows Vista/7/8/10 (32 або 64 біти), система моделювання Scilab 6.0.0.

Завантаження програми здійснюється шляхом відкриття файлу VB_AppDurEst.sce стандартними засобами ОС.

ДОДАТОК Г – Інструкція користувача

Загальний опис роботи з програмою

Після завантаження файлу VB_AppDurEst.sce на екрані з'являється форма для відображення початкового стану програми для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК (рисунок Г.1).

Рисунок Г.1 – Форма для відображення початкового стану програми

Далі у вікні редагування «Файл з даними» графічного вікна «Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК»

(рис.Г.1) потрібно відредагувати ім'я текстового файлу з початковими емпіричними даними, що будуть використовуватися для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

Після натискання на командну кнопку «Ввести» з текстового файлу вводяться дані (попарно значення тривалості розробки у місяцях та трудомісткості у людино-годинах), які виводяться у окремому графічному вікні «Емпіричні дані».

Потім у графічному вікні «Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК» (рис. Г.1) у відповідних вікнах редагування потрібно відредагувати: трудомісткість розробки у людино-годинах та довірчу ймовірність.

Крім того у графічному вікні «Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК» (рис. Г.1) потрібно за допомогою радіо кнопки здійснити вибір наявності фактичного розміру ПЗ та ввести його для подальшого порівняння з відповідною оцінкою тривалості розробки у місяцях.

Далі потрібно натиснути на командну кнопку «Оцінювання». Після цього з'являться результати оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК.

І на останок у графічному вікні «Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК» головного модуля програми (рис. Г.1) потрібно натиснути на командну кнопку «Закрити». Це приведе до закриття всіх графічних вікон та завершення програми.

ДОДАТОК Д – Програма та методика випробувань

Об’єкт випробувань: Програма для оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК VB_AppDurEst.

Ціль випробувань: перевірка відповідності програми VB_AppDurEst до вимог технічного завдання і повноти реалізації.

Склад програмної документації:

- технічне завдання;
- інструкція користувача;
- опис програми;
- текст програми;
- програма та методика випробувань.

Технічні вимоги.

Програма має експлуатуватись на обладнанні з параметрами не нижчими за наступні: процесор Pentium IV (або еквівалентний) з інструкціями SSE2, оперативна пам’ять 2 Гб (1 Гб мінімум), простір на жорсткому диску 600 Мб мінімум.

Методи випробувань.

Випробування програми виконується представником замовника в присутності представника розробника. Відповідно до опису програми виконується запуск програми VB_AppDurEst. Далі виконується перевірка роботи програми. Виконується перевірка всіх екранних форм програми, а також всіх допустимих для виконання операцій всередині екранних форм згідно до посібника користувача. Якщо програма демонструє адекватне функціонування згідно до вимог ТЗ та представленої програмної документації, то робота з

розробки програми вважається виконаною, і виконується впровадження програми до експлуатації. В протилежному випадку програма має бути відправлено на доробку розробникові.

Використовуються тести за методологією „чорної скриньки” (рис.Д.1-Д.4).

Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

File Tools Edit ?

Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Дані для оцінювання		Файл з даними	Наявність фактичної тривалості у місяцях
Трудомісткість розробки (людино-год.)	512	ata_F\WB_Duraton_39_2.txt	☒ є (включити), ☐ немає (виключити)
Довірча ймовірність	0.95	Ввести	0
Оцінювання			
Результати оцінювання			
Результат нел. регресії тривалості (місяці)			
Нижня границя довір.інтервалу тривалості (міс.)		Нижня границя інтервалу прогноз. тривалості (міс.)	
Верхня границя довір.інтервалу тривалості (міс.)		Верхня границя інтервалу прогноз. тривалості (міс.)	
Закрити			

Рисунок Д.1 – Графічне вікно головного модуля програми

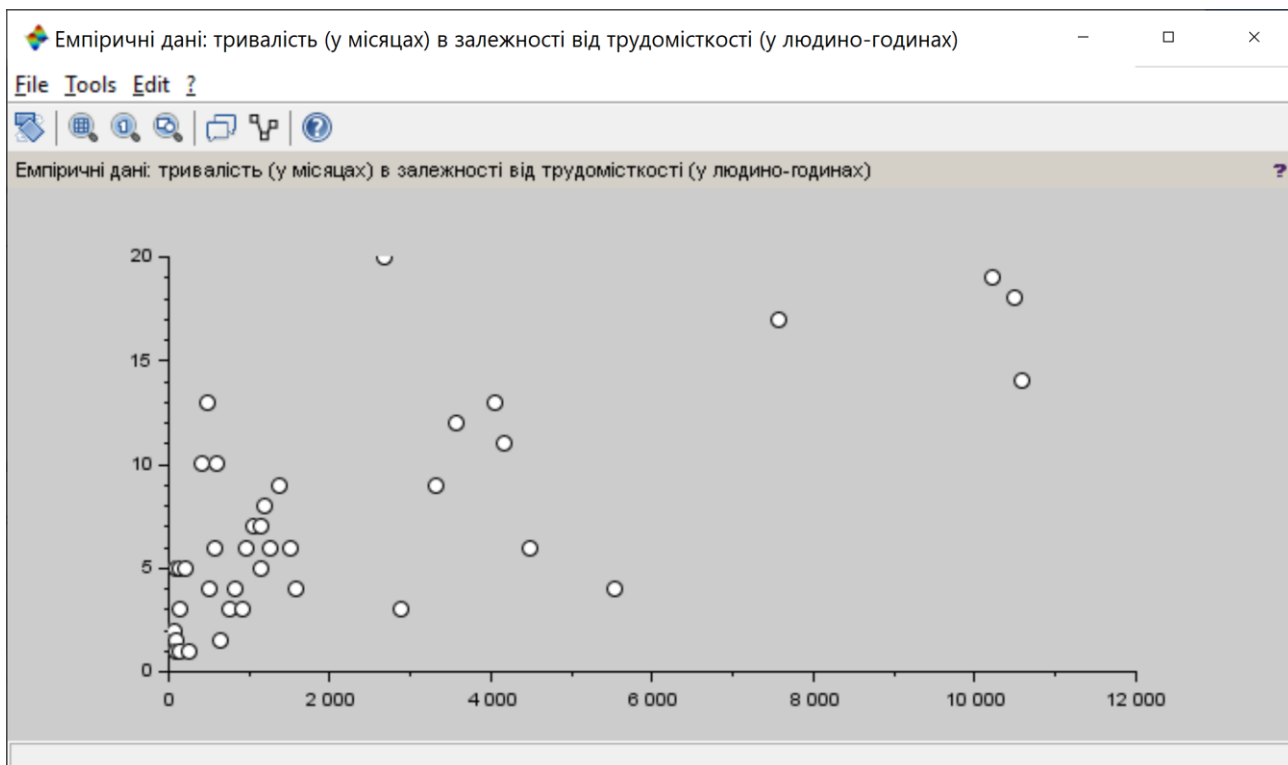


Рисунок Д.2 – Візуалізація тестових емпіричних даних.

Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

File Tools Edit ?

Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Дані для оцінювання		Файл з даними	Наявність фактичної тривалості у місяцях	
Трудомісткість розробки (людино-год.)	80	ata_F\VB_Duraton_39_2.txt	☐ є (включити), ☐ немає (виключити)	
Довірча ймовірність	0.95	Ввести	0	
Оцінювання				
Результати оцінювання		Коефіцієнт детермінації $R^2=0.4824$		
Результат нел. регресії тривалості (місяц)	=2.0259			
Нижня границя довір.інтервалу тривалості (міс.)	=1.3650	Нижня границя інтервалу прогноз. тривалості (міс.)	=0.5614	
Верхня границя довір.інтервалу тривалості (міс.)	=3.0068	Верхня границя інтервалу прогноз. тривалості (міс.)	=7.3109	
Закрити				

Рисунок Д.3 – Результати роботи програми після натискання на командну кнопку «Оцінювання» у разі, коли радіо кнопка виключена

Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

File Tools Edit ?

Оцінювання тривалості розробки застосунків, що створюються мовою Visual Basic для ПК

Дані для оцінювання

Трудомісткість розробки (людино-год.) 80

Довірча ймовірність 0.95

Файл з даними ata_F\VB_Duraton_39_2.txt

Ввести

Наявність фактичної тривалості у місяцях

☒ є (включити), немає (виключити) 2

Оцінювання

Результати оцінювання

Коефіцієнт детермінації $R^2=0.4824$

Відносна похибка MRE=0.0129

Результат нел. регресії тривалості (місящ) =2.0259

Нижня границя довір.інтервалу тривалості (міс.) =1.3650

Верхня границя довір.інтервалу тривалості (міс.) =3.0068

Нижня границя інтервалу прогноз. тривалості (міс.) =0.5614

Верхня границя інтервалу прогноз. тривалості (міс.) =7.3109

Закрити

Рисунок Д.4 – Результати роботи програми після натискання на командну кнопку «Оцінювання» у разі, коли радіо кнопка включена та введена фактична тривалість розробки – це 2 місяці