

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

_____ проф. Приходько С.Б.

«__» _____ 2022 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти «бакалавр»

на тему: Розробка програмного забезпечення для моделювання тривалості створення застосунків на Java для ПК

Виконав: студент групи 4151

Лиско Д.І.

(підпис, ПІБ)

Керівник роботи:

Зав.кафедри ПЗАС, д.т.н., професор

(посада, науковий ступень вчене звання)

Приходько С.Б.

(підпис, ПІБ)

Миколаїв – 2022 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

_____ доц. Макарова Л.М.

(підпис)

«__» _____ 2022 р.

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

Студенту Лиско Дмитру Ігоровичу

(Прізвище, ім'я, по батькові)

1. Тема роботи: Розробка програмного забезпечення для моделювання тривалості створення застосунків на Java для ПК

Керівник роботи Приходько С.Б.

Затверджені наказом ректора №256-уч від «11» травня 2022 р.

2. Термін подання роботи: 10.06.2022 р.

3. Вихідні дані по роботі: _____

4. Перелік питань, що належать до розробки (найменування розділів) _____

Титульний аркуш; Завдання на кваліфікаційну роботу; Анотація (українською та англійською мовами); Зміст; Перелік умовних позначень, символів, одиниць та термінів (при необхідності); Вступ; Опис предметної галузі та постановка задачі; Проект програмного забезпечення (ескізний, технічний та робочий); Результати розробки програмного забезпечення; Розділ з охорони праці; Висновки; Список використаних джерел; Додатки (технічне завдання, текст програми, опис програми, інструкція користувача, програма та методика випробувань програмного забезпечення).

5. Перелік презентаційних матеріалів: Тема роботи, Мета роботи, Методи моделювання тривалості виконання програмних проектів. Нелінійне рівняння регресії для моделювання тривалості створення застосунків на Java для ПК. Діаграма варіантів використання ПЗ, Діаграма діяльності ПЗ, Графічне вікно програми із інтерфейсними компонентами, Емпіричні дані тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах). Візуалізація нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та одного результату моделювання тривалості. Візуалізація нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та трьох результатів моделювання тривалості. Результат обчислення коефіцієнту детермінації нелінійної регресії. Висновки. _____

АНОТАЦІЯ

Кваліфікаційна робота присвячена розробці програми для моделювання тривалості створення застосунків на Java для ПК.

Кваліфікаційна робота викладена на 77 сторінках машинописного тексту, містить 15 рисунків, 3 таблиці, 5 додатків та список використаної літератури з 12 найменувань.

ABSTRACT

The qualification paper is devoted to the development of software for simulation of effort of software projects.

The qualification paper is presented on 77 pages of typewritten text, contains 15 figures, 3 tables, 5 appendices and 12 references.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ	7
ВСТУП	8
1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ДЛЯ МОДЕЛЮВАННЯ ТРИВАЛОСТІ СТВОРЕННЯ ПРОГРАМНИХ ЗАСТОСУНКІВ ТА ПОСТАНОВКА ЗАДАЧІ	10
1.1 Існуючі методи для моделювання тривалості створення програмних застосунків	10
1.2 Постановка задачі	20
2 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ МОДЕЛЮВАННЯ ТРИВАЛОСТІ СТОРЕННЯ ЗАСТОСУНКІВ НА JAVA ДЛЯ ПК	22
2.1 Розробка ескізного проекту програми для моделювання тривалості створення застосунків на Java для ПК	22
2.2 Розробка технічного проекту програми для моделювання тривалості створення застосунків на Java для ПК	28
2.3 Розробка робочого проекту програми для моделювання тривалості створення застосунків на Java для ПК	34
3 РЕЗУЛЬТАТИ РОЗРОБКИ	41
4 ОХОРОНА ПРАЦІ	48
4.1 Аналіз існуючої проблеми охорони праці	48
4.2 Аналіз шкідливих та небезпечних у офісному приміщенні з персональними комп'ютерами	49
4.3 Розробка заходів щодо зменшення впливу небезпечних і шкідливих факторів	53
ВИСНОВКИ	57
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	58
ДОДАТОК А – Технічне завдання на розробку програми для моделювання тривалості створення застосунків на Java для ПК	60

ДОДАТОК Б – Текст програми	63
ДОДАТОК В – Опис програми	69
ДОДАТОК Г – Інструкція користувача	70
ДОДАТОК Д – Програма та методика випробувань	73

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ

ПЗ – програмне забезпечення;

COCOMO – the COnstructive COst MOdel;

ISBSG – International Software Benchmarking Standards Group.

ВСТУП

В галузі інженерії програмного забезпечення (ПЗ) є потреба в оцінюванні тривалості створення програмних застосунків шляхом відповідного моделювання, яке робиться із застосуванням в тому числі нелінійних рівнянь регресії [1-5], наприклад, відомі моделі COCOMO (The Constructive Cost Model) та ISBSG (International Software Benchmarking Standards Group). Враховуючи, що регресійні моделі містять випадкову складову, виникає потреба у відповідному моделюванні залежної змінної. Існуюче ПЗ, яке застосовується для моделювання (наприклад, LabView, MatLab та Scilab), не має у своєму складі відповідних функцій, які дозволяли би це робити. Крім того, Java є популярною мовою програмування, а персональний комп'ютер (ПК) є широко поширеним багатоцільовим комп'ютером. Тому виникає потреба у вирішенні практичної задачі інженерії ПЗ, що характеризується комплексністю та невизначеністю умов, яка полягає у розробці оригінальної програми для моделювання тривалості створення саме застосунків на Java для ПК.

Метою кваліфікаційної роботи є розробка проекту саме такої програми для моделювання тривалості створення застосунків на Java для ПК.

Для досягнення цієї мети необхідно вирішити наступні задачі.

- 1) Виконати аналіз існуючих методів для моделювання тривалості створення застосунків на Java для ПК.
- 2) Зробити постановку задачі на розробку програми для моделювання тривалості створення застосунків на Java для ПК.
- 3) Розробити ескізний проект програми для моделювання тривалості

створення застосунків на Java для ПК.

4) Розробити технічний проект програми для моделювання тривалості створення застосунків на Java для ПК.

5) Розробити робочий проект програми для моделювання тривалості створення застосунків на Java для ПК.

1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ДЛЯ МОДЕЛЮВАННЯ ТРИВАЛОСТІ СТВОРЕННЯ ПРОГРАМНИХ ЗАСТОСУНКІВ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Існуючі методи для моделювання тривалості створення програмних застосунків

В галузі програмної інженерії є потреба в оцінюванні тривалості створення програмних застосунків. шляхом відповідного моделювання, яке робиться із застосуванням в тому числі нелінійних рівнянь регресії [1-5], наприклад, відома модель COCOMO II (The Constructive Cost Model) та ISBSG (International Software Benchmarking Standards Group). Враховуючи, що регресійні моделі містять випадкову складову, виникає потреба у відповідному моделюванні залежної змінної. Існуюче програмне забезпечення (ПЗ), яке застосовується для моделювання (наприклад, LabView, MatLab, Scilab), не має у своєму складі відповідних функцій, які дозволяли це робити. Тому виникає потреба у створенні оригінальної програми для моделювання тривалості створення застосунків на Java для ПК.

Якщо випадкові величини z_X і Z_Y мають нормальний розподіл, то проблем з побудовою рівняння регресії немає тому, що воно буде лінійним

$$\bar{Z}_Y = b_0 + b_1 z_X. \quad (1.1)$$

В загальному випадку рівняння регресії є таким:

$$Z_Y = \bar{Z}_Y + \delta,$$

де δ – випадкова величина з нормальним розподілом.

У разі нормального розподілу не виникає проблем і з побудовою інтервалу прогнозування лінійної регресії. Цей інтервал визначається із використанням квантилів розподілу Стюдента [6]

$$\hat{Z}_{Y_i} \pm t_{\alpha/2, N-2} S_{Z_Y} \sqrt{1 + \frac{1}{N} + \frac{(z_{X_i} - \bar{z}_X)^2}{S_{Z_X Z_X}}}, \quad (1.2)$$

$\hat{Z}_{Y_i}$ значення $\bar{Z}_Y$ за рівнянням (1.1) за значенням z_{X_i} ; z_{X_i} – i -те значення z_X ;

$t_{\alpha/2, N-2}$ – квантіль t -розподілу Стюдента з $N - 2$ степенями свободи та $\alpha/2$

рівнем значимості; $S_{Z_X Z_X} = \sum_{i=1}^N (z_{X_i} - \bar{z}_X)^2$; $\bar{z}_X = \frac{1}{N} \sum_{i=1}^N z_{X_i}$;

$$S_{Z_Y}^2 = \frac{1}{N-1} \sum_{i=1}^N (z_{Y_i} - \bar{z}_Y)^2; \quad \bar{z}_Y = \frac{1}{N} \sum_{i=1}^N z_{Y_i}.$$

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 нелінійного рівняння регресії 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 найкращої моделі тому, що вона в цю сукупність може не потрапити. Із-за цього для побудови нелінійного рівняння регресії прагнуть використовувати інші методи, які не потребують простого перебору. До цих методів відносять методи, що базуються на лінеаризуючих та нормалізуючих перетвореннях. Методи, що використовують лінеаризацію, нажаль можуть застосовуватися лише в поодиноких випадках, коли вдається знайти лінеаризуюче перетворення. І навпаки, нормалізуюче перетворення завжди вдається підібрати, що дає можливість застосовувати для побудови нелінійного рівняння регресії 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 нелінійного рівняння регресії [7].

Аналогічно із застосуванням нормалізуючих перетворень в роботі [8] було запропоновано визначати інтервал прогнозування нелінійної регресії.

Практичне застосування нормалізуючих перетворень для моделювання тривалості створення застосунків на Java для ПК потребує пошуку відповідного перетворення. У якості нормалізуючого перетворення в [8] рекомендується використовувати перетворення Джонсона.

Перетворення Джонсона і методи оцінювання його параметрів

Ідея перетворення однієї випадкової величини в іншу, яка має більш просту щільність розподілу (в першу чергу нормальну), була розвинена Джонсоном. У 1949 році Джонсон представив нормалізуюче перетворення з трьох сімей, яким відповідають певні розподіли. Сім'ї розподілів Джонсона отримані шляхом перетворення нормованої нормально розподіленої випадкової величини z з нульовим математичним сподіванням і одиничною дисперсією. В загальному випадку перетворення має вигляд [9-11]

$$z = \gamma + \eta q(x, \varphi, \lambda); \quad \eta > 0; \quad -\infty < \gamma < \infty; \quad \lambda > 0; \quad -\infty < \varphi < \infty, \quad (1.3)$$

де q – довільна функція; γ , η , λ , φ – параметри розподілу Джонсона. Слід відзначити, що γ і η – це параметри форми, λ – це параметр масштабу, параметр φ характеризує зсув розподілу у напрямку осі абсцис.

Джонсон запропонував три різні сім'ї функцій q :

$$1) \quad q(x, \varphi, \lambda) = \ln \left(\frac{x - \varphi}{\lambda} \right), \quad x > \varphi \text{ (сім'я } S_L);$$

$$2) \quad q(x, \varphi, \lambda) = \ln \left(\frac{x - \varphi}{\lambda + \varphi - x} \right), \quad \varphi < x < \varphi + \lambda \text{ (сім'я } S_B);$$

$$3) \quad q(x, \varphi, \lambda) = \operatorname{Arsh} \left(\frac{x - \varphi}{\lambda} \right), \quad -\infty \leq x \leq +\infty \text{ (сім'я } S_U),$$

$$\text{де } \operatorname{Arsh} \left(\frac{x - \varphi}{\lambda} \right) = \ln \left[\frac{x - \varphi}{\lambda} + \sqrt{\left(\frac{x - \varphi}{\lambda} \right)^2 + 1} \right].$$

Для сім'ї S_L функція щільності ймовірності задається як

$$f_L(x) = \frac{\eta}{\sqrt{2\pi}(x-\varphi)} \exp\left\{-\frac{\eta^2}{2} \left[\frac{\gamma - \eta \ln \lambda}{\eta} + \ln(x-\varphi) \right]^2\right\},$$

де $x > \varphi$; $\eta > 0$; $-\infty < \gamma < \infty$; $\lambda > 0$; $-\infty < \varphi < \infty$.

Ця функція щільності ймовірності зростає від φ до точки максимуму і потім більш повільно спадає у разі спрямування x у нескінченність ($x \rightarrow \infty$). Функція щільності ймовірності для сім'ї S_L завжди унімодальна, асиметрія завжди додатна, а ексцес більший за 3 (значення ексцесу нормального розподілу).

Для сім'ї S_B функція щільності ймовірності задається формулою

$$f_B(x) = \frac{\eta\lambda}{\sqrt{2\pi}(x-\varphi)(\lambda+\varphi-x)} \exp\left\{-\frac{1}{2} \left[\gamma + \eta \ln\left(\frac{x-\varphi}{\lambda+\varphi-x}\right) \right]^2\right\},$$

де $\varphi < x < \varphi + \lambda$; $\eta > 0$; $-\infty < \gamma < \infty$; $\lambda > 0$; $-\infty < \varphi < \infty$.

Функції щільності ймовірності сім'ї S_B можуть бути як унімодальними, так і бімодальними. Необхідні та достатні умови для бімодальності полягають у тому, що

$$\eta < 1/\sqrt{2}, \quad |\gamma| < \eta^{-1} \sqrt{1-2\eta^2} - 2\eta \operatorname{arth} \sqrt{1-2\eta^2}.$$

Для сім'ї S_U функція щільності ймовірності визначається як

$$f_U(x) = \frac{\eta}{\sqrt{2\pi\{(x-\varphi)^2 + \lambda^2\}}} \exp\left\{-\frac{1}{2} \left[\gamma + \eta \ln\left(\frac{x-\varphi}{\lambda} + \sqrt{\left(\frac{x-\varphi}{\lambda}\right)^2 + 1}\right) \right]^2\right\},$$

де $-\infty < x < \infty$; $\eta > 0$; $-\infty < \gamma < \infty$; $\lambda > 0$; $-\infty < \varphi < \infty$.

Графіки функції щільності ймовірності сім'ї S_U мають високий порядок стику з віссю абсцис на кінцях інтервалу зміни аргументу, є унімодальними та їх моди лежать між медіаною та нулем. Тім самим ці розподіли мають додатну або від'ємну асиметрію в залежності від того, від'ємна або додатна величина параметра γ .

Сім'ї розподілів Джонсона відрізняються багатостатністю форм і у площині асиметрії у квадраті A^2 та ексцесу ε займають значні області. На рисунку 1.1 представлена діаграма Джонсона (області комбінацій A^2 і ε для різних розподілів Джонсона). Ця діаграма дозволяє підібрати сім'ю розподілів Джонсона за значеннями оцінок A^2 та ε вибіркового розподілу.

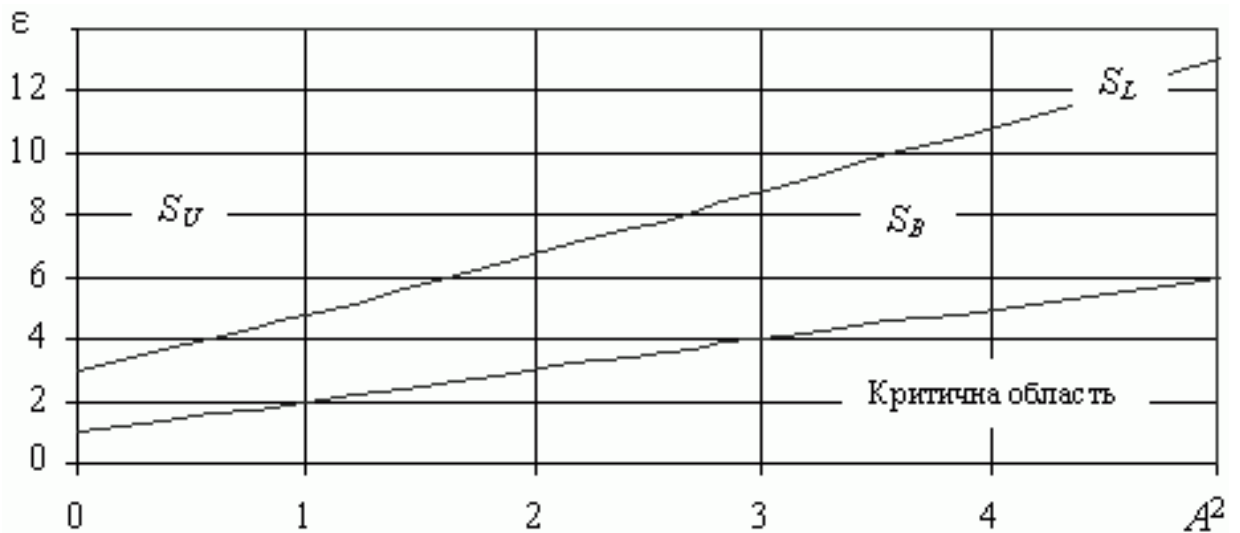


Рисунок 1.1 – Комбінації A^2 і ε для різних розподілів Джонсона

Якщо комбінація A^2 і ε знаходиться біля лінії S_L , то вибіркового розподілу можна апроксимувати розподілом із сім'ї S_L (логарифмічно нормальним розподілом). Розподіл зі значеннями A^2 і ε , які лежать вище лінії S_L , апроксимуються розподілом із сім'ї S_U , а які лежать нижче лінії S_L (до лінії критичної області) – розподілом із сім'ї S_B . Якщо комбінація A^2 і ε попадає в критичну область, то вибіркового розподілу не можна апроксимувати розподілом із сімей Джонсона.

Практично завжди при підборі аналітичної моделі закону розподілу експериментальних даних асиметрія A та ексцес ε не бувають відомі. У цьому випадку вибір сім'ї розподілів Джонсона здійснюється за оцінками асиметрії A та ексцесу ε , які знаходяться за гистограмою. Після вибору необхідної сім'ї розподілів Джонсона обчислюють його параметри, і виконують перевірку

адекватності обраної моделі експериментальним даним за критерієм згоди, наприклад, критерієм Пірсона (χ^2 -критерієм).

Ексцес ε визначається як

$$\varepsilon = \mu_4(x)/\sigma_x^4,$$

де $\mu_4(x)$ – центральний статистичний момент четвертого порядку випадкової величини X ; σ_x – середнє квадратичне відхилення випадкової величини X .

Асиметрія A визначається як

$$A = \mu_3(x)/\sigma_x^3,$$

де $\mu_3(x)$ – центральний статистичний момент третього порядку випадкової величини X .

Лінія верхньої границі критичної області задається залежністю

$$\varepsilon = A^2 + 1.$$

Лінію S_L в межах $A^2 \in [0, 5]$ можна апроксимувати наступною функцією:

$$\varepsilon = 0,0428A^4 + 1,7948A^2 + 2,9953.$$

Лінію S_L в межах $A^2 \in [0, 27]$ можна апроксимувати наступною функцією [12]:

$$\varepsilon(A^2) = 7,2315 \cdot 10^{-6} A^8 - 6,9860 \cdot 10^{-4} A^6 + 4,5460 \cdot 10^{-2} A^4 + 1,7979 A^2 + 2,9891.$$

За цими залежностями можна автоматизувати вибір нормалізуючого перетворення Джонсона певної сім'ї.

Параметри функції щільності ймовірності γ , η , λ та ϕ для обраної сім'ї розподілів Джонсона в загальному випадку можна знайти шляхом рішення наступної задачі математичного програмування:

$$\hat{\theta} = \arg \min_{\theta} \left\{ \sum_{j=1}^m [y(x_j) - f(x_j, \theta)]^2 \right\}, \quad (1.4)$$

де θ – вектор невідомих параметрів, $\theta = \{\gamma, \eta, \lambda, \phi\}$; $\hat{\theta}$ – оцінка вектора θ ; x_j – значення випадкової величини x в середині j -го підінтервалу; $y(x_j)$ –

значення ординати гістограми при значенні x_j ; $f(x_j, \theta)$ – вираз функції щільності ймовірності при значенні x_j ; m – кількість підінтервалів гістограми. Зазначимо, що при використанні формули (1.4), у якості значення $y(x_j)$ потрібно брати відношення відносної частоти в j -му підінтервалі до довжини j -го підінтервалу Δx .

Параметри функції щільності ймовірності γ , η , λ та ϕ для обраної сім'ї розподілів Джонсона також можна знайти із рішення системи рівнянь, яка складена з рівностей відповідних аналітичних виразів для статистичних моментів $\alpha_1(x)$, $\mu_2(x)$, $\mu_3(x)$ і $\mu_4(x)$ та їх оцінок, обчислених за вибірковими даними гістограми.

Відповідні оцінки статистичних моментів $\alpha_1(x)$, $\mu_2(x)$, $\mu_3(x)$ і $\mu_4(x)$ знаходяться за гістограмою за наступними формулами:

$$\hat{\alpha}_1(x) = \bar{x} = \sum_{j=1}^m y(x_j) \Delta x x_j;$$

$$\hat{\mu}_k(x) = \sum_{j=1}^m y(x_j) \Delta x [x_j - \bar{x}]^k, \quad k = 2, 3, 4.$$

Вважається, що застосування перетворення Джонсона потребує побудови аналітичної моделі емпіричного розподілу випадкової величини у вигляді відповідної щільності ймовірності Джонсона, яка залежить від чотирьох параметрів. А це в свою чергу призводить до необхідності мати велику вибірку значень випадкової величини. Виникає питання: що робити у разі малої вибірки? В цьому разі перетворення Джонсона також може бути застосовано.

Для вирішення проблеми підбору нормалізуючого перетворення Джонсона у разі малої вибірки в роботі [11] запропоновано використовувати непараметричний підхід, суть якого полягає у наступному. Спочатку за вибіркою значень випадкової величини знаходять оцінки ексцесу та асиметрії, за якими за діаграмою Джонсона обирають певну сім'ю: S_L , S_B або S_U . Далі

в залежності від обраної сім'ї визначають параметри перетворення Джонсона в результаті рішення наступної задачі [11]:

$$\hat{\theta} = \arg \min_{\theta} \{A_z^2 + (\varepsilon_z - 3)^2\}, \quad (1.5)$$

де $\hat{\theta}$ – оцінка вектора невідомих параметрів θ , $\theta = \{\gamma, \eta, \lambda, \phi\}$ для перетворення

$$(1.3); A_z = \frac{1}{nS_z^3} \sum_{i=1}^n (z_i - \bar{z})^3; \bar{z} = \frac{1}{n} \sum_{i=1}^n z_i; \varepsilon_z = \frac{1}{nS_z^4} \sum_{i=1}^n (z_i - \bar{z})^4; S_z^2 = \frac{1}{n} \sum_{i=1}^n (z_i - \bar{z})^2;$$

z_i – i -те значення нормалізованої випадкової величини z у вибірці довжиною n , $i \in [1, n]$, визначаються за перетвореннями (1.3).

Перевірку адекватності побудованої аналітичної моделі експериментальним даним зазвичай здійснюють за критерієм згоди, наприклад, критерієм Пірсона (χ^2 -критерієм).

Відповідно за цим критерієм спочатку обчислюють значення χ^2

$$\chi^2 = \sum_{j=1}^m \frac{(n_j - np_j)^2}{np_j},$$

де m – кількість підінтервалів (кліток), на які розбивається інтервал $[x_{\min}, x_{\max}]$; $x_{\min}$ і $x_{\max}$ – відповідно мінімальне і максимальне значення випадкової величини X ; n_j – абсолютна частота в j -му підінтервалі (кількість значень випадкової величини, які попадають у j -ий підінтервал); p_j – ймовірність того, що значення випадкової величини X попадають у j -ий підінтервал.

Кількість підінтервалів (кліток), на які розбивається інтервал $[x_{\min}, x_{\max}]$ може бути визначений за наступними формулами: $m = \log_2 n + 1 = 3,3 \lg n + 1$ (формула Старджеса) або $m = 5 \lg n$ (формула Брукса і Карузера).

Ймовірність того, що значення випадкової величини X попадають у j -ий підінтервал можуть бути визначені як

$$p_j = \int_{x_{j-1}}^{x_j} f(x) dx,$$

де x_{j-1} і x_j – ліва і права границі j -го підінтервалу.

Після цього в залежності від рівня значимості α та кількості ступенів вільності ν за таблицею верхніх 100α %-вих точок розподілу χ^2 знаходять значення $\chi_{кр}^2$. Якщо $\chi^2 \leq \chi_{кр}^2$, то з ймовірністю $1-\alpha$ можна прийняти гіпотезу про те, що закон розподілу результатів спостережень відповідає нульовій гіпотезі, якщо $\chi^2 > \chi_{кр}^2$ – цю гіпотезу потрібно відкинути.

Кількість ступенів вільності ν визначається як

$$\nu = m - k - 1,$$

де k – кількість параметрів, від яких залежить закон розподілу. Для нормального розподілу $k = 2$, для розподілів Джонсона $k = 4$.

Для перевірки адекватності нелінійного рівняння регресії використаємо коефіцієнт детермінації R^2 :

$$R^2 = 1 - \left(\sum_{i=1}^n (y_i - \hat{y}_i)^2 / \sum_{i=1}^n (y_i - \bar{y})^2 \right),$$

де: y_i – i -те значення y за емпіричними даними, $\hat{y}_i$ – i -те значення y , яке розраховане за рівнянням регресії, $\bar{y}$ – середнє значення y , яке обчислене за емпіричними даними.

Розглянемо побудову рівняння нелінійної регресії за емпіричними даними, який в подальшому використаємо для моделювання тривалості створення застосунків на Java для ПК. За даними про тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) з [5] було побудовано нелінійне рівняння регресії за методом на основі нормалізуючих перетворень. Ці дані були взяті з репозитарію ISBSG релізу 2021 року та наведені на рис.1.2.

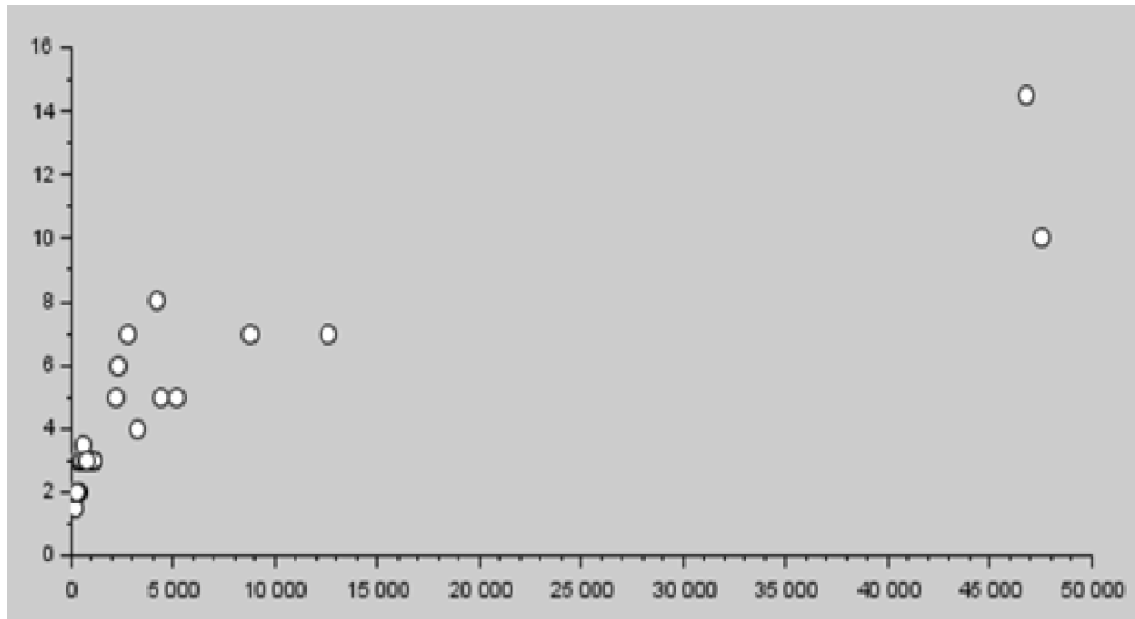


Рисунок 1.2 – Дані тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) з репозитарію ISBSG релізу 2021 року

У якості нормалізуючого перетворення було обрано перетворення Джонсона (1.3) із сім'ї S_B . За нормалізованими даними було побудовано лінійне рівняння регресії (1.1), де відповідні коефіцієнти дорівнювали: $b_0 = 0$; $b_1 = 0,9424$.

Після підстановки в лінійне рівняння регресії (1.1) перетворення Джонсона (1.3) із сім'ї S_B було отримано рівняння нелінійної регресії

$$\bar{y} = \frac{e^{c_1(\lambda_y + \varphi_y)} + \varphi_y}{1 + e^{c_1}}, \quad (1.6)$$

де $c_1 = \frac{1}{\eta_y} \cdot (b_1 \cdot z_x + b_0 - \gamma_y)$; $z_x = \gamma_x + \eta_x \ln\left(\frac{x - \varphi_x}{\lambda_x + \varphi_x - x}\right)$; параметри

перетворення Джонсона γ_y , η_y , φ_y , λ_y , γ_x , η_x , φ_x та λ_x мають відповідно значення 7,20870, 1,5329788, 0,7037417, 301,669344, 4,3573688, 0,5912588, 67,50, та 1342643,20.

В подальшому це нелінійне рівняння регресії (1.6) використаємо для моделювання тривалості створення застосунків на Java для ПК.

1.2 Постановка задачі

Виходячи з аналізу існуючих методів для моделювання тривалості створення застосунків на Java для ПК, з опису перетворення Джонсона та методів оцінювання його параметрів, сформулюємо постановку задачі. Потрібно створити програму для моделювання тривалості створення застосунків на Java 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 (значення випадкових величин).
- 2) Здійснювати вибір нормалізуючого перетворення Джонсона.
- 3) Виконувати нормалізацію емпіричних даних за вибраним перетворенням Джонсона.
- 4) Знаходити значення коефіцієнтів лінійного рівняння регресії за нормалізованими емпіричними даними.
- 5) Здійснювати побудову нелінійного рівняння регресії.
- 6) Обчислювати значення коефіцієнта детермінації R^2 .
- 7) Здійснювати побудову інтервалу прогнозування нелінійної регресії.
- 8) Виводити на екран нелінійну регресію, інтервал прогнозування нелінійної регресії та результати для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

9) Виводити у текстовий файл результати для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

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.

2 РОЗРОБКА ПРОЕКТУ ПРОГРАМИ ДЛЯ МОДЕЛЮВАННЯ ТРИВАЛОСТІ СТВОРЕННЯ ЗАСТОСУНКІВ НА JAVA ДЛЯ ПК

2.1 Розробка ескізного проекту програми для моделювання тривалості створення застосунків на Java для ПК.

В ескізному проекті було виявлено варіанти використання та побудовано їх сценарії, розроблено зовнішню специфікацію програми для моделювання тривалості розробки застосунків, що створюються мовою Java 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 2.1.

Таблиця 2.1

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.
Користувач	Ввести трудомісткість розробки ПЗ	Вводить трудомісткість (у людино-годинах) розробки програмного застосунку, що створюється мовою Java для ПК, за яким потрібно змодельовати тривалість (у місяцях) розробки програмного застосунку, що створюється мовою Java для ПК

Продовження табл. 2.1

1	2	3
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 Java 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 Java для ПК
Користувач	Змодельовати тривалість	Отримує довірчий інтервал та інтервал прогнозування нелінійної регресії тривалості розробки програмних застосунків, що створюються мовою Java для ПК, із заданою довірчою ймовірністю та моделює тривалість

Діаграма варіантів використання програми для моделювання тривалості розробки програмних застосунків, що створюються мовою Java для ПК, зображена на рисунку 2.1.

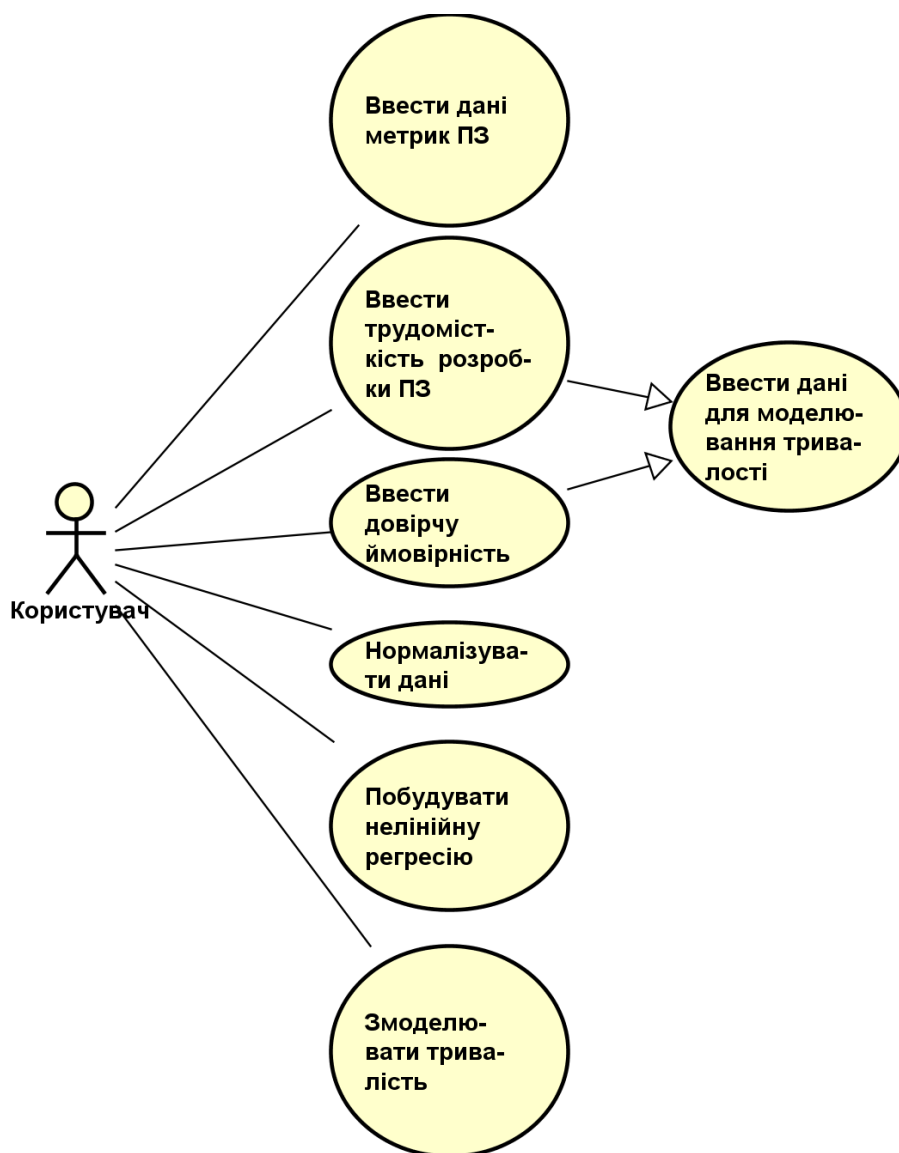


Рисунок 2.1 – Діаграма варіантів використання ПЗ для моделювання тривалості розробки застосунків, що створюються мовою Java для ПК

Сценарії варіантів використання зображені у вигляді діаграми діяльності на рисунку 2.2.

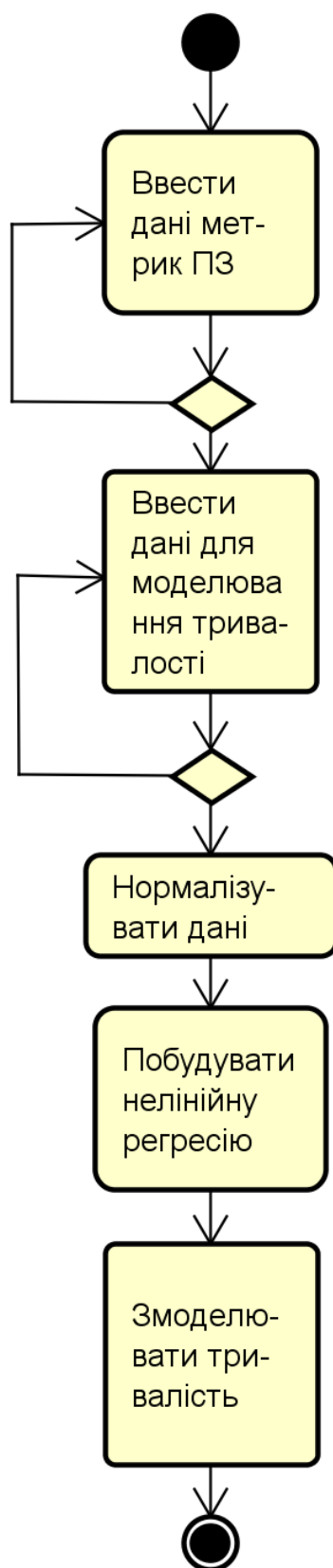


Рисунок 2.2 – Діаграма діяльності ПЗ для моделювання тривалості розробки застосунків, що створюються мовою Java для ПК

Зовнішня специфікація програми.

Назва програми: SoftwareDurSim.

Функції, які виконує програма:

- 1) Вводити з текстового файлу емпіричні дані (значення випадкових величин).
- 2) Здійснювало вибір нормалізуючого перетворення Джонсона.
- 3) Виконувало нормалізацію емпіричних даних за вибраним перетворенням Джонсона.
- 4) Знаходило значення коефіцієнтів лінійного рівняння регресії за нормалізованими емпіричними даними.
- 5) Здійснювало побудову нелінійного рівняння регресії.
- 6) Обчислювало значення коефіцієнта детермінації R^2 .
- 7) Здійснювати побудову інтервалу прогнозування нелінійної регресії.
- 8) Виводити на екран нелінійну регресію, інтервал прогнозування нелінійної регресії та результати моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) .
- 9) Виводити у текстовий файл результати моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Вхідні дані:

- значення випадкової величини (емпіричні дані): тривалість (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах);
- сім'я нормалізуючого перетворення Джонсона;
- значення параметрів нормалізуючого перетворення Джонсона;
- трудомісткість (у людино-годинах), в залежності від якої потрібно змоделювати тривалість (у місяцях) створення застосунків на Java для ПК;
- кількість значень тривалості створення застосунків на Java для ПК, що потрібно змоделювати.

Вихідні дані:

- нормалізовані емпіричні дані;
- нелінійна регресія, інтервал прогнозування нелінійної регресії та результати для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Зовнішні ефекти: поява графічного вікна для вводу даних, управління виводом даних, графічного вікна для виводу нелінійної регресії, інтервалу прогнозування нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах), вихід з програми (закриття всіх графічних вікон).

Для проектування інтерфейсу програми для моделювання тривалості створення застосунків на Java для ПК представимо стани програми при виконанні у вигляді відповідної діаграми станів. Діаграма станів програми для моделювання тривалості створення застосунків на Java для ПК наведена на рисунку 2.3.

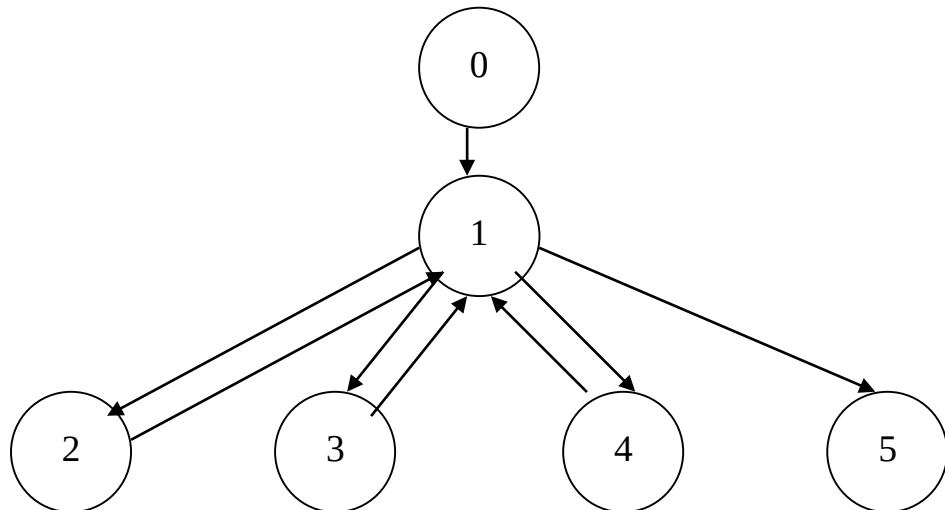


Рисунок 2.3 – Діаграма станів програми для моделювання тривалості створення застосунків на Java для ПК

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 рівняння нелінійної регресії, 4 – графічне вікно з результатами моделювання тривалості створення застосунків на Java для ПК, 5 – кінцевий стан.

2.2 Розробка технічного проекту програми для моделювання тривалості створення застосунків на Java для ПК

В технічному проекті було спроектовано програму для моделювання тривалості створення застосунків на Java для ПК на основі покрокової деталізації зверху вниз, розроблено специфікації модулів програми, побудовано алгоритми модулів програми.

Програма складається з таких модулів: головного модуля (SoftwareDurSim), модуля для введення емпіричних даних з файлу та їх візуалізації (DataInput), модуля для побудови нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) (Software_Dur_Sim), модуля для виводу нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості у людино-годинах (Output_yr), модуля для закриття графічних вікон (_Close). Схема взаємодії модулів програми наведена на рисунку 2.4.

Специфікація головного модуля.

Назва модуля: SoftwareDurSim.

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 нормалізуючого перетворення Джонсона певної сім'ї.

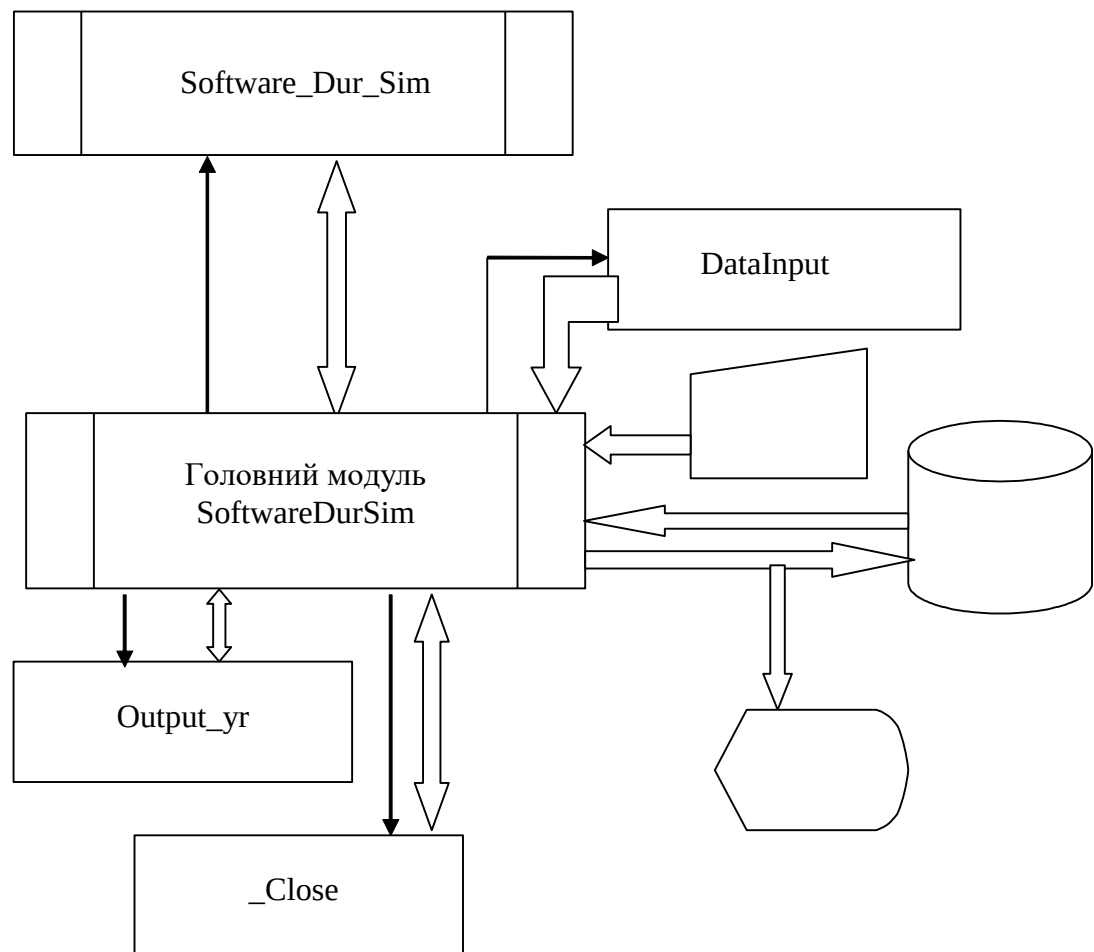


Рисунок 2.4 – Схема взаємодії модулів програми

5) Виклик функції для визначення нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

6) Виклик функції для виводу нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

7) Виклик функції для закриття графічних вікон.

Вхідні дані:

- сім'я нормалізуючого перетворення Джонсона;
- значення параметрів нормалізуючого перетворення Джонсона;
- розмір програми (у функціональних точках), в залежності від якої потрібно змоделювати трудомісткість (у годинах) виконання програмного проекту;
- кількість значень трудомісткості (у годинах) виконання програмного проекту, що потрібно змоделювати.

Вихідні дані:

- значення нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Зовнішні ефекти: поява графічного вікна для вводу даних, управління виводом даних, моделювання тривалості створення застосунків на Java для ПК., виводу нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) , виходу 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 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 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 побудови нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) .

Назва модуля: Software_Dur_Sim.

Функції, які виконує модуль:

- 1) Визначає нелінійну регресію, інтервал прогнозування нелінійної регресії та результати моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).
- 2) Обчислює значення коефіцієнту детермінації для нелінійної регресії.
- 3) Формує одномірні масиви значень нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів моделювання

тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Вхідні дані:

- сім'я нормалізуючого перетворення Джонсона;
- значення параметрів нормалізуючого перетворення Джонсона;
- розмір програми (у функціональних точках), в залежності від якої потрібно змодельовати трудомісткість (у годинах) виконання програмного проекту;
- кількість значень трудомісткості (у годинах) виконання програмного проекту, що потрібно змодельовати.

Вихідні дані:

- масиви значень нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах);
- значення коефіцієнту детермінації для нелінійної регресії.

Зовнішні ефекти: або з'являється текстове повідомлення «Моделювання виконано за нелінійним рівнянням регресії» та виводиться значення коефіцієнту детермінації для нелінійної регресії, або з'являється текстове повідомлення «Нелінійне рівняння регресії не знайдено, моделювання не виконано».

Специфікація модуля для виводу нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Назва модуля: Output_уг.

Функції, які виконує модуль:

- 1) Відкриття графічного вікна для візуалізації нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для

моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

2) Візуалізацію нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Вхідні дані:

- масиви значень випадкових величини (емпіричних даних);
- масиви значень нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Вихідні дані:

- ідентифікатор графічного вікна для візуалізації нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Зовнішні ефекти: Поява графічного вікна для візуалізації нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

Специфікація модуля для закриття графічних вікон.

Назва модуля: _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.

2.3 Розробка робочого проекту програми для моделювання тривалості створення застосунків на Java для ПК

В робочому проекті було розроблено програму для моделювання тривалості створення застосунків на Java для ПК, здійснено тестування програми, розроблено програмну документацію, здійснено випробування програми.

Спочатку scі-мовою для системи моделювання Scilab було виконано кодування алгоритмів, що були розроблені на стадії технічного проекту. Текст програми наведений в додатку Б. Приклад кодування модуля Software_Dur_Sim, що єдійснює побудову нелінійної регресії та інтервалу прогнозування значень нелінійної регресії, scі-мовою в системі моделювання Scilab наведено нижче.

```
//Функція моделювання тривалості створення застосунків на Java
для ПК
function [yr, piyr_min, piyr_max, r_2, xi, yi, xs,
ys]=Software_Dur_Sim(x, y, n)
//Считываем значение переменных из текстовых полей и
//преобразовываем их к числовому типу.
ax=eval(get(edit_ax,'string'));
bx=eval(get(edit_bx,'string'));
cx=eval(get(edit_cx,'string'));
dx=eval(get(edit_dx,'string'));
ay=eval(get(edit_ay,'string'));
by=eval(get(edit_by,'string'));
cy=eval(get(edit_cy,'string'));
dy=eval(get(edit_dy,'string'));

//Проверяем значение флажка, если флажок выключен,
if (get(radius_bikv_x,'value')==0)&(get(radius_bikv_y,'value')==0)
    zxA=0;
    for i=1:n
        zx(i)=ax+bx*log((x(i)-cx)/(dx-x(i)+cx));
        zy(i)=ay+by*log((y(i)-cy)/(dy-y(i)+cy));
        zxA=zxA+zx(i);
    end;
    zxA=zxA/n;
    S2zx=0;
    for i=1:n
```

```

        S2zx=S2zx+(zx(i)-zxA)^2;
    end;

    b0=0; b1=0.9424;
    //    tn=2.01;

    //    Df=n-1;
    Df=n-2;
    P=0.975;
    Q=1-P;
    //    Df=n-2;
    //    Q=0.05/2/n;
    //    P=1-Q;

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

    //    Delta_z=T/n^0.5;
    Delta_z=T*(1+1/n)^0.5;

    for i=1:n
        zr(i)=b1*zx(i);
        yr(i)=((cy+dy)*exp((zr(i)-ay)/by)+cy)/(1+exp((zr(i)-ay)/by));

        Delta_z=T*(1+1/n+(zx(i)-zxA)^2/S2zx)^0.5;
        pizrmin=zr(i)-Delta_z;
        pizrmax=zr(i)+Delta_z;

        piyr_min(i)=((cy+dy)*exp((pizrmin-ay)/by)+cy)/(1+exp((pizrmin-ay)/by));
        piyr_max(i)=((cy+dy)*exp((pizrmax-ay)/by)+cy)/(1+exp((pizrmax-ay)/by));

    end;

    //set(textresult,'string','Нелінійне рівняння регресії знайдено');
    y_mean=0;
    for i=1:n
        y_mean=y_mean+y(i);
    end;
    y_mean=y_mean/n;
    sy_t=0;
    sy=0;
    for i=1:n
        sy=sy+(y(i)-y_mean)^2;
        sy_t=sy_t+(y(i)-yr(i))^2;;
    end;
    r_2=1-sy_t/sy;
    set(textresult,'string',...
    sprintf("Моделювання виконано за нелінійним рівнянням регресії з \tR^2=%1.4f",r_2));
    else

```

```

    set(textresult,'string',...
        'Нелінійне рівняння регресії не знайдено, моделювання не
        виконано');

end;

//
n_m=900;
Delta_z=T*(1+1/n)^0.5;

for i=1:n_m
    xi(i)=55+50*i;
    zxi=ax+bx*log((xi(i)-cx)/(dx-xi(i)+cx));
    zri=b1*zxi;
    yr(i)=((cy+dy)*exp((zri-ay)/by)+cy)/(1+exp((zri-
ay)/by));
    yi(i)=yr(i);

    Delta_z=T*(1+1/n+(zxi-zxA)^2/S2zx)^0.5;
    pizrmin=zri-Delta_z;
    pizrmax=zri+Delta_z;

    piyr_min(i)=((cy+dy)*exp((pizrmin-
ay)/by)+cy)/(1+exp((pizrmin-ay)/by));
    piyr_max(i)=((cy+dy)*exp((pizrmax-
ay)/by)+cy)/(1+exp((pizrmax-ay)/by));

end;

effort=eval(get(edit_effort,'string'));
k=eval(get(edit_k,'string'));
// k=3;
r = grand(k, 1, "unf", -1, 1);
[xtan]=tan(0.57*r);
[zd]=0.532275*asinh(3.874868*xtan);
for i=1:k
    xs(i)=effort;
    zxi=ax+bx*log((xs(i)-cx)/(dx-xs(i)+cx));
    zri=b1*zxi+zd(i);
    ys(i)=((cy+dy)*exp((zri-ay)/by)+cy)/(1+exp((zri-ay)/by)),
end;

endfunction

```

Далі в середовищі Scilab 6.0.0 було створено ссе-файл розміром 11 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.2.5.

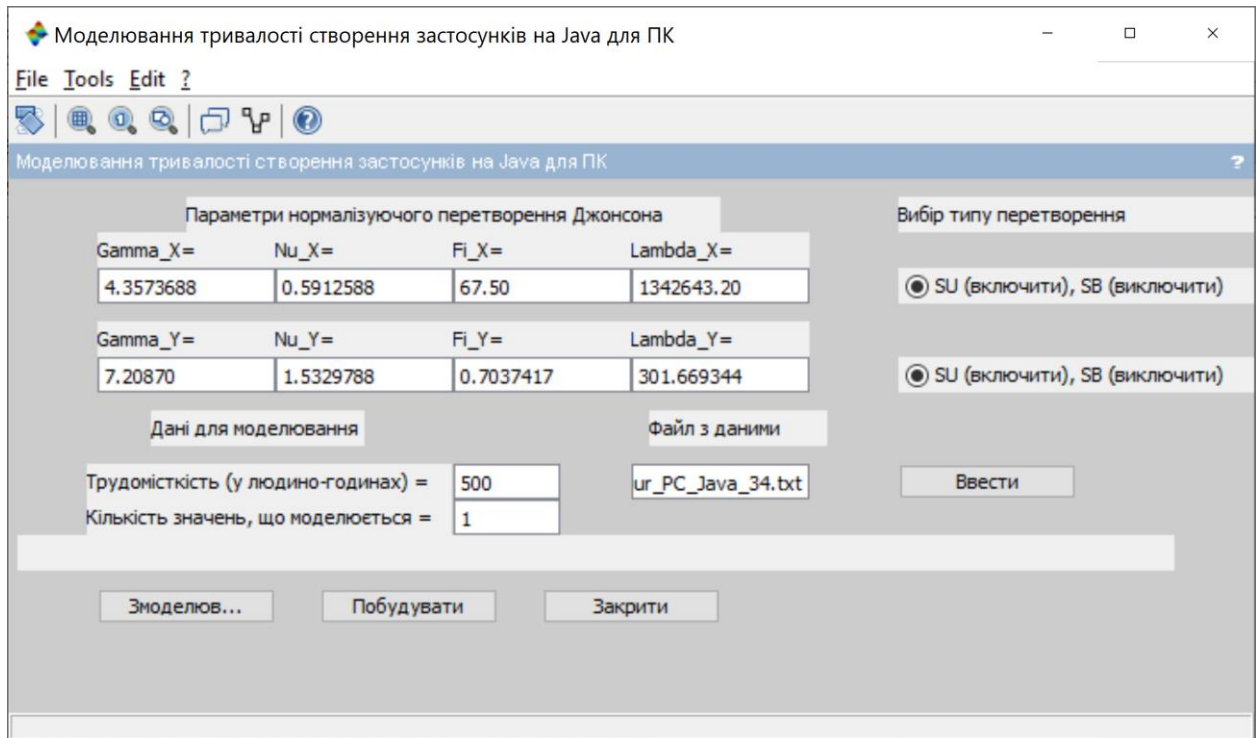


Рисунок 2.5 – Графічне вікно програми із інтерфейсними компонентами

Було проведено тестування окремих модулів. У якості прикладу наведемо результати тестування для модуля моделювання тривалості створення застосунків на Java для ПК за емпіричними даними з тривалості створення 34 застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) з [5]. Було побудовано нелінійне рівняння регресії на основі нормалізуючого перетворення Джонсона сім'ї S_B . Результати візуалізації емпіричних даних з тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) наведені на рис.2.6, візуалізації нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-

годинах) – на рис.2.7, результати обчислення коефіцієнту детермінації R^2 – на рис.2.8.

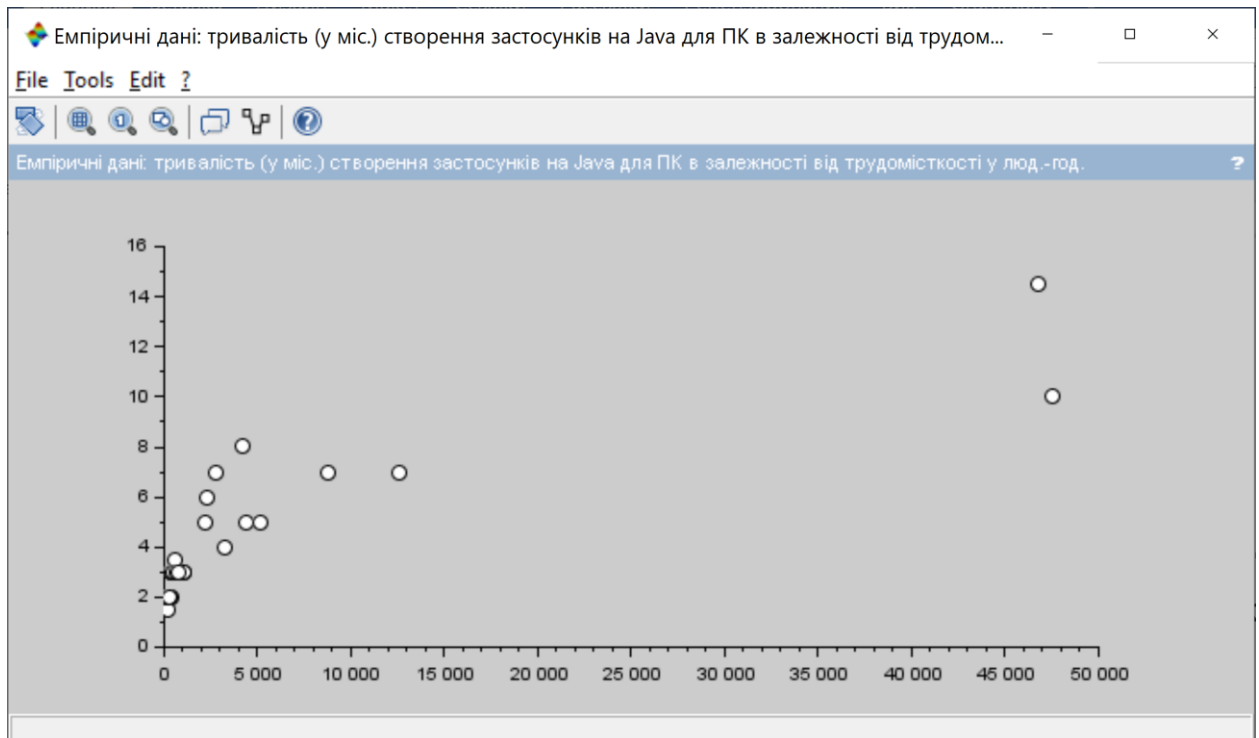


Рисунок 2.6 – Емпіричні дані тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах)

На рис. 2.7 нанесено у вигляді точки результати моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості, що дорівнює 5000 людино-годин. Точка попала майже всередину інтервалу прогнозування значень нелінійної регресії (червоні лінії) трохи вище нелінійної регресії (чорна лінія).

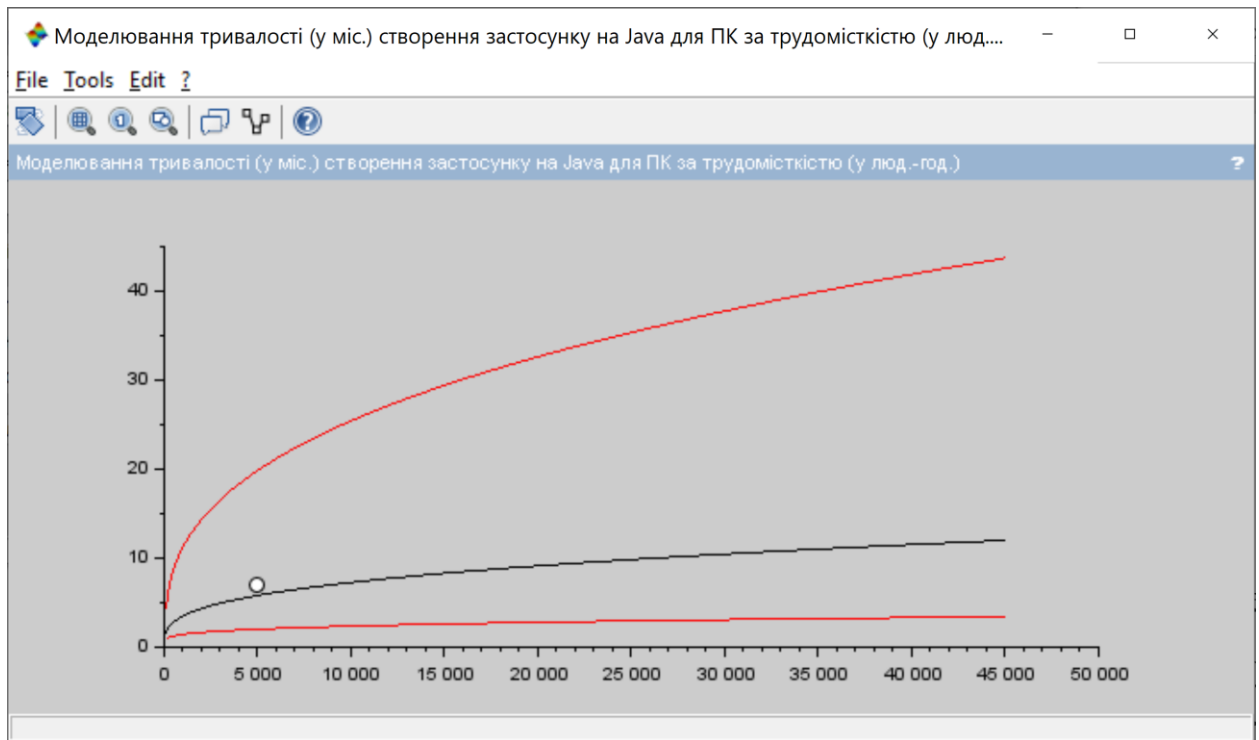


Рисунок 2.7 – Візуалізація нелінійної регресії, інтервалу прогнозування значень нелінійної регресії та результатів для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах)

Моделювання тривалості створення застосунків на Java для ПК

File Tools Edit ?

Моделювання тривалості створення застосунків на Java для ПК

Параметри нормалізуючого перетворення Джонсона

Gamma_X=	Nu_X=	Fi_X=	Lambda_X=
4.3573688	0.5912588	67.50	1342643.20
Gamma_Y=	Nu_Y=	Fi_Y=	Lambda_Y=
7.20870	1.5329788	0.7037417	301.669344

Вибір типу перетворення

☐ SU (включити), SB (виключити)

☐ SU (включити), SB (виключити)

Дані для моделювання

Файл з даними

Трудомісткість (у людино-годинах) = 5000

Кількість значень, що моделюється = 1

ur_PC_Java_34.txt

Ввести

Моделювання виконано за нелінійним рівнянням регресії з $R^2=0.8807$

Змоделюв... Побудувати Закрити

Рисунок 2.8 – Результати обчислення коефіцієнту детермінації R^2 нелінійної регресії тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах)

Отримані результати моделювання, побудови нелінійної регресії тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах) та обчислення коефіцієнту детермінації R^2 практично співпали з результатами роботи [5]. Це свідчить про працездатність розробленої програми для моделювання тривалості створення застосунків на Java для ПК.

3 РЕЗУЛЬТАТИ РОЗРОБКИ

В результаті розробки було створено програму для моделювання тривалості створення застосунків на Java для ПК для системи моделювання Scilab 6.0.0. Назва програми: SoftwareDurSim. В середовищі Scilab 6.0.0 було створено sce-файл SoftwareDurSim розміром 11 КБ. Проведено випробування програми.

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. Результати випробувань програми наведені на рис. 3.1-3.5. При випробуванні програми вводилися із текстового файлу наступні тестові дані:

```
4372 5
385 2
2786 7
429 3
408 3
46787 14.5
2330 6
210 2
694 3
355 3
535 3.5
489 3
5175 5
12563 7
245 2
491 3
8772 7
```

600	3
3213	4
606	3
175	2
1018	3
509	3
167	1.5
899	3
105	2
47493	10
4224	8
147	2
669	3
2158	5
1007	3
784	3
245	2

Після запуску програми відкривається графічне вікно «Модельовання тривалості створення застосунків на Java для ПК» головного модуля програми (рис. 3.1). Далі у вікні редагування «Файл з даними» цього графічного вікна потрібно відредагувати ім'я текстового файлу з початковими емпіричними даними, що будуть використовуватися для модельовання тривалості створення застосунків на Java для ПК.

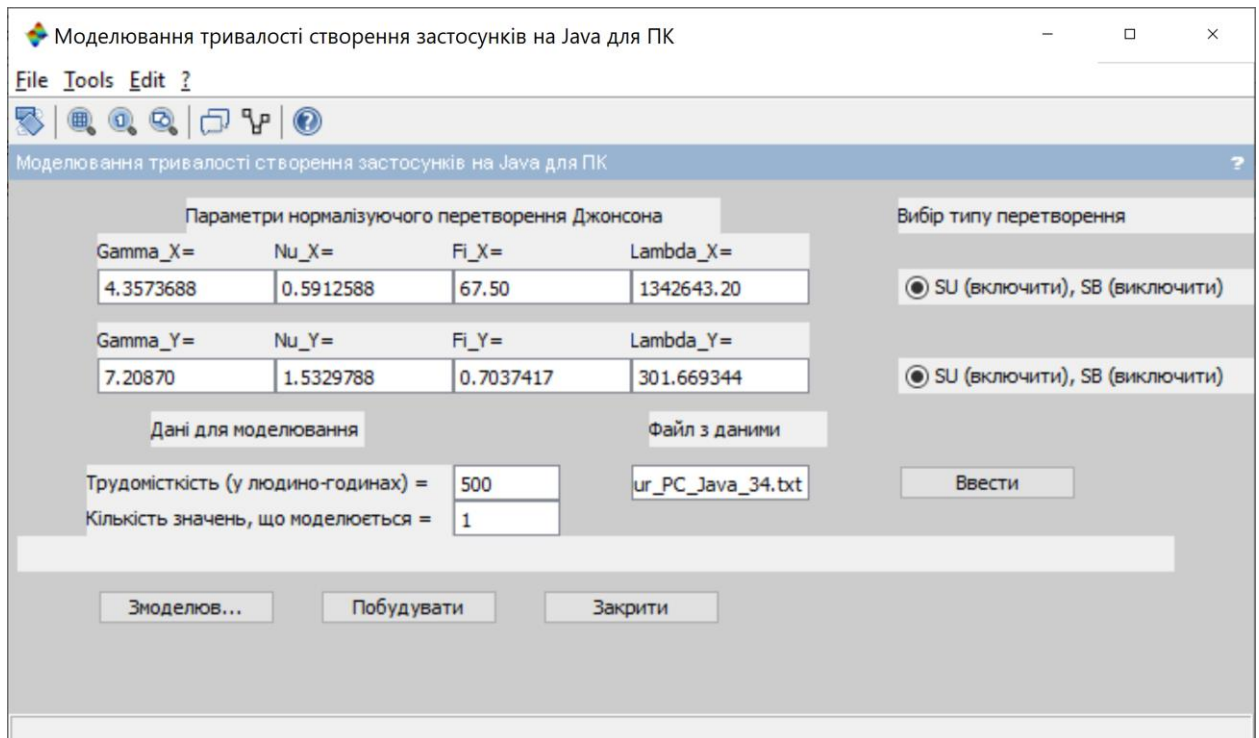
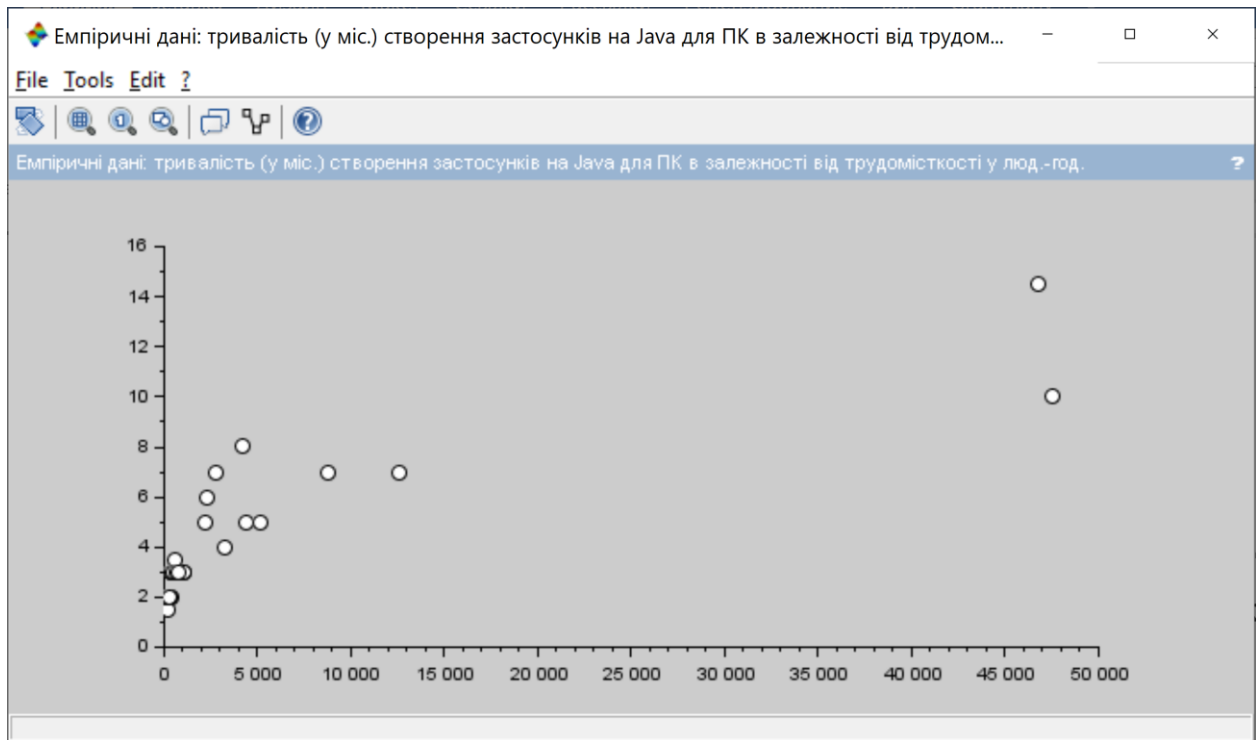


Рисунок 3.1 – Графічне вікно головного модуля програми

Після натискання на командну кнопку «Ввести» з текстового файлу вводяться дані (попарно значення трудомісткості у людино-годинах та тривалості створення застосунків на Java для ПК у місяцях), які виводяться у окремому графічному вікні «Емпіричні дані», наведено на рис. 3.2.



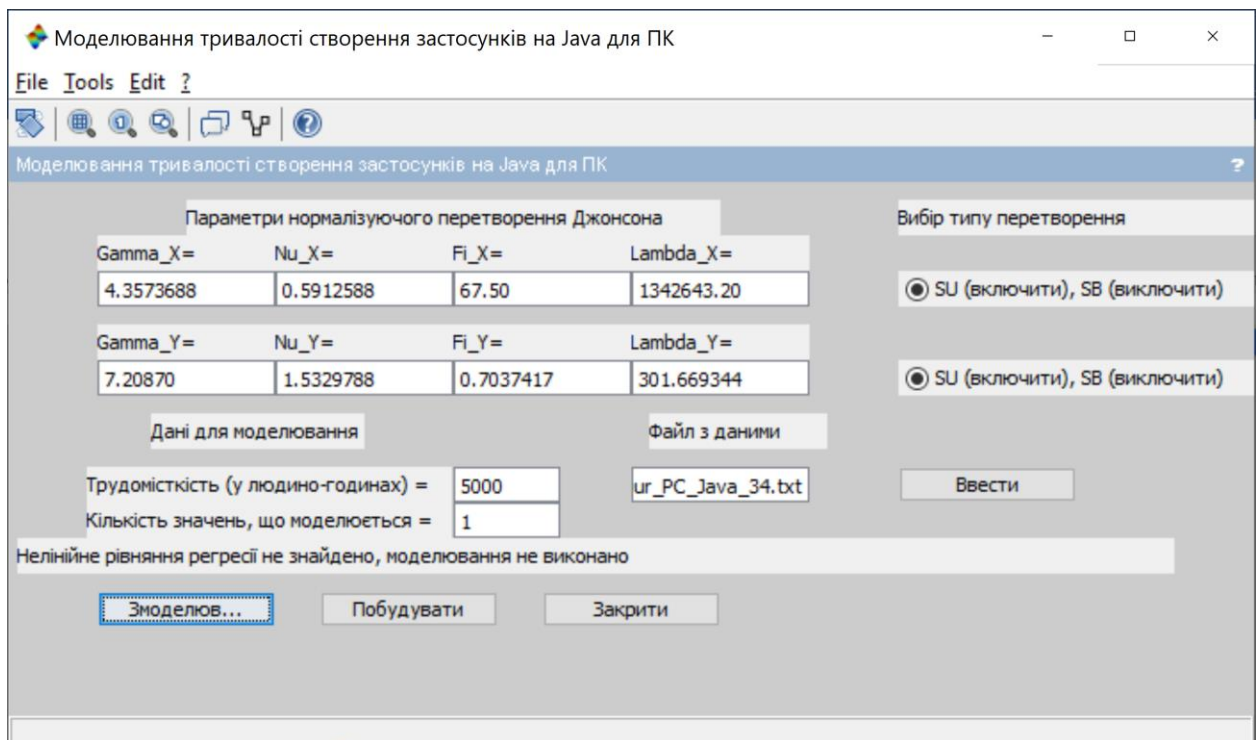


Рисунок 3.3 – Результати роботи програми після помилкового вибору типу (сім'ї) перетворення Джонсона і редагування його параметрів

Після цього знову у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» головного модуля програми (рис. 3.1) потрібно за допомогою двох радіо кнопок здійснити правильний вибір типу (сім'ї) перетворення Джонсона і коректне редагування його параметрів у відповідних вікнах редагування «Параметри нормалізуючого перетворення Джонсона». Далі натиснути на командну кнопку «Змоделювати». Якщо все було зроблено правильно, то у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» з'являється текст «Моделювання виконано за нелінійним рівнянням регресії» і виводиться значення коефіцієнту детермінації (рис. 3.4).

Рисунок 3.4 – Результат обчислення коефіцієнту детермінації
нелінійної регресії

Після цього у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» головного модуля програми (рис. 3.1) потрібно натиснути на командну кнопку «Побудувати». Це приведе до появи графічного вікна «Моделювання тривалості (у міс.) створення застосунків на Java для ПК за трудомісткістю (у люд.-год.)», яке наведено на рис. 3.5. На рис. 3.5 нанесено у вигляді 3 точок результати для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від їх трудомісткості, що дорівнює 5000 людино-годин. Точки попали майже всередину інтервалу прогнозування значень нелінійної регресії (червоні лінії) біля нелінійної регресії (чорна лінія).

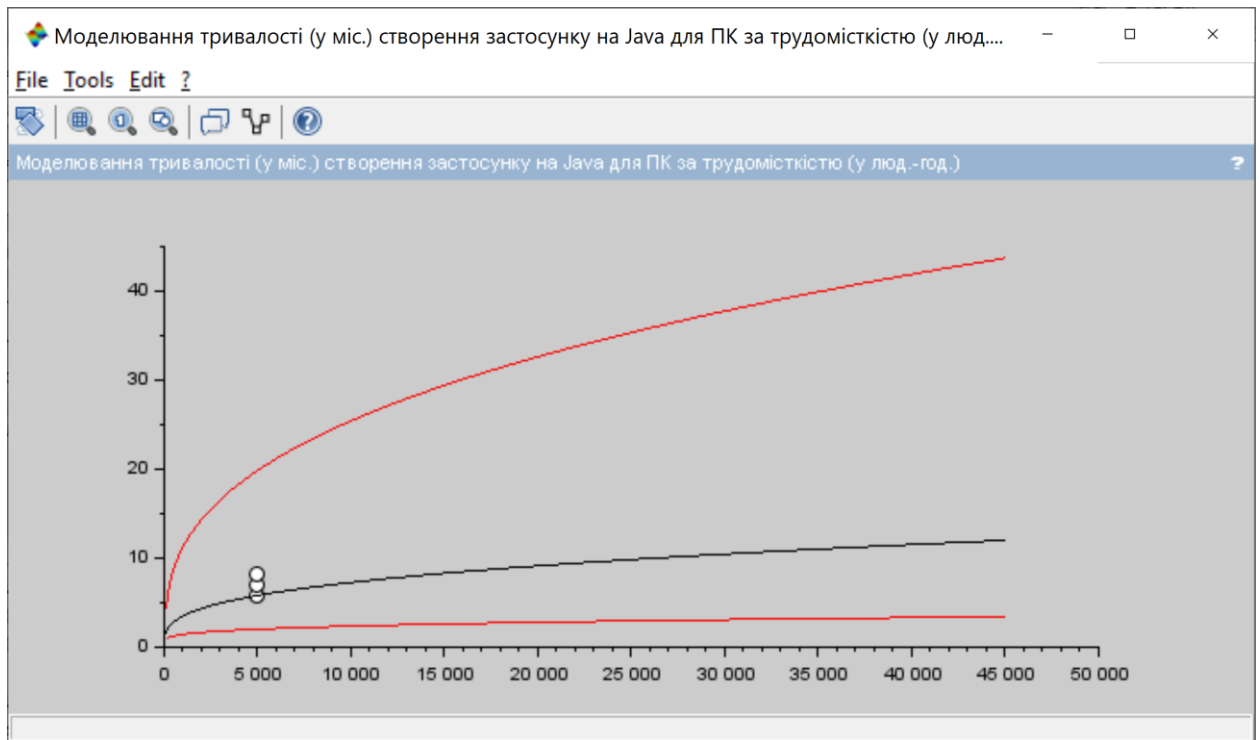


Рисунок 3.5 – Графічне вікно «Моделювання тривалості (у міс.) створення застосунків на Java для ПК за трудомісткістю (у люд.-год.)» при трудомісткості, що дорівнює 5000 люд.-год.

І на останок у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» головного модуля програми (рис. 3.1) потрібно натиснути на командну кнопку «Закрити». Це приведе до закриття всіх графічних вікон та завершення програми.

За результатами випробувань зроблено висновок про працездатність розробленої програми для моделювання тривалості створення застосунків на Java для ПК для системи моделювання Scilab 6.0.0.

4 ОХОРОНА ПРАЦІ

4.1 Аналіз існуючої проблеми охорони праці

Охорона праці – це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних та лікувально-профілактичних заходів та засобів, спрямованих на збереження здоров'я і працездатності людей в процесі праці (ст. 1 Закону «Про охорону праці»). Як правовий інститут охорона праці – це чималий комплекс правових норм. Можна визначити охорону праці також як систему забезпечення безпеки, збереження здоров'я і працездатності людини в процесі праці, засновану на сукупності законодавчих актів і відповідних їм соціально-економічних, технічних, гігієнічних і організаційних заходів.

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.

4.2 Аналіз шкідливих та небезпечних у офісному приміщенні з персональними комп'ютерами

Небезпечним називається виробничий фактор, вплив якого на працюючу людину в певних умовах приведе до травми або іншому раптовому різкому погіршенню здоров'я. Якщо ж виробничий фактор приведе до захворювання або зниження працездатності, то його вважають шкідливим. Залежно від рівня й тривалості впливу шкідливий виробничий фактор може стати небезпечним.

На офісного працівника при роботі впливають дві основні групи шкідливих факторів:

1) Фізичні, до яких відносять фактори, що породжуються безпосереднім впливом оточуючого середовища (електромагнітне випромінювання, освітленість робочого місця, температура повітря, шум).

2) Психофізіологічні, до яких відносять фізичне та нервово-психічне перевантаження організму (зорове напруження, статичні навантаження, нервово-психічне навантаження).

Освітлення – отримання, розподіл та використання світлової енергії для забезпечення нормальних умов праці. Світло, крім забезпечення фізичної можливості розрізняти предмети праці, також впливає на психічний стан людини та перебіг фізіологічних процесів в організмі. Раціонально влаштоване освітлення виробничих приміщень позитивно впливає на працівників, підвищує ефективність та безпеку праці, знижує втому та травматизм, забезпечує високу працездатність.

За вимогами санітарних норм освітлення повинно бути достатньо яскравим, рівномірним, не засліплювати очі та не створювати відблисків на

робочій поверхні. За спектральним складом освітлення має наближатися до сонячного світла. Гігієнічними нормами вимагається максимально використовувати природне освітлення, оскільки денне світло краще сприймається органами зору. Штучне освітлення у виробничих та адміністративно-громадських приміщеннях має здійснюватись системою загального рівномірного освітлення, допускають застосування системи комбінованого освітлення. Значення освітленості на поверхні робочого столу в зоні розміщення документів має становити 300-500 лк. Як джерела світла для штучного освітлення мають застосовувати переважно люмінесцентні лампи типу ЛБ.

Одним з найнесприятливіших факторів, що зменшує працездатність та уважність робітників, створює передумови до травматизму та захворювань, є шум. Інтенсивний шум та вібрації негативно впливають не лише на слух, а і на нервову, серцево-судинну, та більшість інших систем організму, викликаючи зниження слуху, гіпертонію, психічні розлади тощо.

Гігієнічними нормативами (СН 3222-85, ГОСТ 12.1.003-85, ГР 2411-81) встановлені допустимі рівні звуку різних частот для робітників певних професій. У таблиці 4.1 наведені нормативи для людей, що працюють з ЕОМ.

Виходячи з даних норм, при проведенні комплексу заходів зі зменшення шуму усувається не будь-яка віброакустична активність обладнання, а тільки та, яка перевищує встановлений граничний рівень та шкідливо впливає на організм людини. Усунення промислового шуму в першу чергу відбувається шляхом вдосконалення технологічних процесів та обладнання. При цьому в першу чергу усуваються найбільш сильні джерела шуму. Також необхідно знешумлювати усе обладнання, яке цього потребує, оскільки знешумлення окремих одиниць при наявності великої кількості обладнання, що продовжує випромінювати шум, недоцільно.

Таблиця 4.1

Допустимі рівні шуму

Вид трудової діяльності	Рівні звукового тиску в дБ октавних смугах із середньо геометричними частотами, ГЦ									Рівні звуку, еквівалентні рівні звуку, дБа/дБаекв
	31,5	63	125	250	500	1000	2000	4000	8000	
Програмісти ЕОМ	86	71	61	54	49	45	42	40	38	50
Оператори в залах обробки інформації на ЕОМ та оператори комп'ютерного набору	96	83	74	68	63	60	57	55	54	65
В приміщеннях для розташування шумних агрегатів ЕОМ	103	91	83	77	73	70	68	66	64	75

Самопочуття та працездатність людини також сильно залежать від теплового стану організму, на який впливають такі фізичні фактори виробничого середовища, як температура повітря, вологість, швидкість руху повітря, атмосферний тиск, доля кисню у повітрі тощо. Наприклад, якщо кількість кисню в повітрі зменшиться до 12% то утруднюється дихання. В таких умовах людина напружує дихальний апарат, дихає частіше; такий стан людина витримує до 0,5 години. Перегрівання організму відбувається за умов надлишкового конвективного випромінювання тепла нагрітих поверхонь. При перегріванні вступають в активну реакцію пристосувальні функції організму. При цьому активізується робота серцево-судинної та дихальної систем, відбувається інтенсивне потовиділення, котре сягає 5 л за зміну. З потом втрачається велика кількість мінеральних солей та вітамінів. Вологість повітря суттєво впливає на терморегуляцію людського організму. Підвищення відносної вологості повітря у виробничому приміщенні ускладнює терморегуляцію,

зменшення тепловиділення організмом. Фізіологічно оптимальною є відносна вологість в межах 40..60%.

Сукупність описаних показників стану навколишнього середовища називають мікрокліматом. Для приміщень з ЕОМ встановлені норми мікроклімату приведені у таблиці 4.2.

Таблиця 4.2

Норми мікроклімату для приміщень з ЕОМ

Пора року	Категорія робіт	Температура повітря, С, не більше	Відносна вологість повітря, %	Швидкість руху повітря, м/с
Холодна	Легка – 1а	22-24	40-60	0,1
	Легка – 1б	21-23	40-60	0,1
Тепла	Легка – 1а	23-25	40-60	0,1
	Легка – 1б	22-24	40-60	0,2

Робота комп'ютерів і периферійних пристроїв можлива завдяки електричному струмові, до джерел якого вони підключаються, тому існує небезпека ураження користувача струмом. Щоб уникнути нещасних випадків варто строго дотримуватися правил електробезпеки.

Небезпека електричного струму на відміну від інших небезпек збільшується тим, що людина не в змозі без спеціальних приладів виявити напругу дистанційно, а також швидкоплинністю поразки - небезпека виявляється, коли людина вже уражена. Ураження струмом приводить до різних порушень в організмі, викликаючи як місцеву поразку тканин і органів, так і загальне ураження організму.

Людина відчуває дратівну дію змінного струму промислової частоти силою 0,6 -1,5 мА і постійного струму 5-7 мА. Ці струми не представляють серйозної небезпеки для організму людини, тому що при такій силі струму можливе самостійне звільнення людини від контакту зі струмоведучими частинами. Для перемінного струму промислової частоти сила невідпускаючого струму знаходиться в межах 6-20 мА. Постійний струм не

викликає невідпускаючого ефекту, але приводить до сильних болючих відчуттів, сила такого струму 15-80 мА і більше.

4.3 Розробка заходів щодо зменшення впливу небезпечних та шкідливих факторів

Сьогодні фахівці в області ергономіки вже зрозуміли, що не можна знайти ідеальне положення, у якому можна перебувати і працювати протягом усього робочого дня. Для більшості людей комфортабельним робочим місцем повинне бути таке, котре можна пристосувати не менш чим для двох позицій, при цьому положення крісла, монітора повинні щораз відповідати виконуваній роботі і звичкам. Багато хто вважають, що для роботи на комп'ютері більше підходять вертикальне і злегка похиле положення. Можливо, щоб крісло було злегка нахилене вперед.

Схеми розміщення робочих місць із персональним комп'ютером повинні ураховувати відстані між робочими столами з відеомоніторами (в напрямку тилу поверхні одного відеомонітора й екрана іншого відеомонітора), які повинні бути не менш 2 м, а відстань між бічними поверхнями відеомоніторів – не менш 1,2 м.

Робочий стіл повинен мати простір для ніг висотою не менш 600 мм, шириною не менш 500 мм, глибиною на рівні колін не менш 450 мм і на рівні витягнутих ніг не менш 650 мм.

Робоче місце повинне бути обладнане підставкою для ніг, що має ширину не менш 300 мм, глибину не менш 400 мм, регулювання по висоті в межах 150 мм і по куті нахилу опорної поверхні підставки до 20°. Поверхня підставки повинна бути рифленою й мати по передньому краї бортик висотою 10 мм.

Дисплей. Положення тіла звичайно відповідає напрямку погляду; дисплеї, розташовані занадто низько чи під неправильним кутом, є основними причинами сутулості. Відстань від дисплея до очей може варіюватися в залежності від характеру виконуваної роботи – 40-70 см. При роботі з текстом відстань від екрана до очей повинна лише небагато перевищувати відстань між книгою й очима і складати 40-45 див. Необхідно давати відпочинок очам і час від часу їх просто закривати. Не можна допускати, щоб очі увесь час були сфокусовані на одній відстані, працюючи з текстом, рекомендується установлювати великий шрифт. Дуже важливо, щоб екран монітора не мерехтів. Частота регенерації зображення повинна бути не менш 72 Гц, тому що при більш низькій частоті помітне мерехтіння, хоча люди з особливо чутливим зором можуть помітити його і при більш високій частоті. Їм доведеться підібрати графічну плату і монітор з частотою регенерації 85 Гц і вище.

Рекомендується встановлювати на екран монітора спеціальні скляні поляризаційні фільтри з заземленням. Необхідно уникати того, щоб термінал був звернений екраном убік вікна, оскільки інтенсивна освітленість області зору може затопити потоками світла очі і розмити зображення на сітківці. Якщо приходить сидіти поруч з вікном, то треба розташуватися під прямим кутом до нього, причому екран дисплея повинний бути перпендикулярний шибці – цим виключаються відблиски на екрані.

Для зниження впливу низькочастотного електромагнітного випромінювання монітора на ЕЛТ рекомендується вибирати монітор, що задовольняє стандарту MPR II чи TCO 92-95. У протилежному випадку необхідно установити на екран захисний фільтр, найбільш сучасні моделі, затримують до 99% електромагнітного випромінювання.

Форма спинки крісла повинна повторювати форму спини працюючого. Крісло необхідно установити на такій висоті, щоб не почувався тиск на куприк (крісло розташоване занадто низько) на стегна (крісло розташоване занадто високо). Фахівці з ергономіки, вважали, що кут між стегнами і хребтом

повинний складати 90° , однак недавно проведені дослідження показали, що більшість людей переважно сидять трохи відкинувшись.

Клавіатуру необхідно установити так, щоб не треба було до неї тягтися і нахилитися вперед. При зміні положення тіла (наприклад, з вертикального на похиле) обов'язково треба перемінити положення клавіатури і дисплея. Може виявитися корисною регульована підставка клавіатури, завдяки якій можна без усякої напруги працювати з маніпулятором "миша". Можна поставити клавіатуру і на коліна. Руки при роботі повинні бути прямими в зап'ястях і зігнуті в ліктях приблизно під прямим кутом. Пальці також повинні бути злегка зігнуті. Удари по клавішах не повинні бути занадто сильними.

Зручна висота столу особливо важлива в тому випадку, коли на ньому розташовується клавіатура. Якщо в клавіатури немає підставки, а висоту столу не можна змінити (і він занадто високий), то треба вище підняти сидіння крісла, а під ноги підставити лавочку: чи що-небудь інше. Якщо стіл занадто низький, необхідно підкласти що-небудь під його ніжки.

Якщо при роботі часто приходиться дивитися на документи, треба установити підставку з оригіналом документа в одній площині і на одній, висоті з екраном. Якщо треба частіше дивитися на оригінал, чим на екран, необхідно повернути крісло й екран таким чином, щоб оригінал розміщався прямо перед очима, а комп'ютер ледве збоку.

Щогодини необхідно робити перерву в роботі. У цей час рекомендується робити вправи для зап'ясть. Нижче описаний можливий комплекс вправ.

- Покласти руку на край столу долонею вниз. Взявшись, за пальці, іншою рукою відвести кисть назад і утримувати протягом 5 секунд. Повторити для іншої руки.
- Злегка упертися рукою в стіл і на 5 секунд напружити пальці в зап'ястя. Повторити іншою рукою.
- Сильно зжати пальці, а потім розпрямити їх.

Виконання приведених рекомендацій дозволяє скоротити вплив шкідливих виробничих факторів на організм людини.

ВИСНОВКИ

В кваліфікаційній роботі розроблено програму для моделювання тривалості створення застосунків на Java для ПК.

В роботі були вирішені наступні задачі.

- 1) Проаналізовано існуючі методи для моделювання тривалості створення застосунків на Java для ПК.
- 2) Проаналізовано існуючі методи вибору перетворень для нормалізації емпіричних даних.
- 3) Зроблено постановку задачі на розробку програми для моделювання тривалості створення застосунків на Java для ПК.
- 4) Розроблено ескізний проект програми для моделювання тривалості створення застосунків на Java для ПК.
- 5) Розроблено технічний проект програми для моделювання тривалості створення застосунків на Java для ПК.
- 6) Розроблено робочий проект програми для моделювання тривалості створення застосунків на Java для ПК.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Boehm, B. W. Software engineering economics [Text] / B. W. Boehm. – Englewood Cliffs, NJ : Prentice Hall, 1981. – 768 p. – ISBN 0-13-822122-7.
2. Oligny, S. Exploring the relation between effort and duration in software engineering projects [Text] / S. Oligny, P. Bourque, A. Abran, B. Fournier // In proc. of the World Computer Congress, Aug. 2000. – P.175-178.
3. Приходько, С.Б. Розробка нелінійної регресійної моделі тривалості програмних проєктів на основі нормалізуючого перетворення Джонсона [Текст] / С. Б. Приходько, А.В.Пухалевич // Радіoeлектронні і комп'ютерні системи. – Харків, 2012. – № 4 (56) – С.90-93.
4. Prykhodko, S.B. Developing PC Software Project Duration Model based on Johnson transformation [Text] / S. B. Prykhodko, A. V. Pukhalevich // Proceedings of the 12th International Conference Modern Problems of Radio Engineering, Telecommunications and Computer Science TCSET'2014, Lviv-Slavske, Ukraine. – Lviv: Polytechnic National University, 2014. – pp.114-116.
5. 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/wp-content/uploads/sites/4/NUOS-ITMAS-2021-Proceedings.pdf>
6. Chandola, V. Anomaly Detection: A Survey [Text] / V. Chandola, A. Banerjee, V. Kumar // Technical Report 07-017, 2007. – 72 p.
7. Приходько, С.Б. Метод побудови нелінійних рівнянь регресії на основі нормалізуючих перетворень [Текст]: тези доп. міждерж. наук.-методич. конф. / С. Б. Приходько // Проблеми математичного моделювання. – Дніпродзержинськ: ДДТУ, 2012. – С. 31–33.

8. Prykhodko, S.B. Statistical anomaly detection technique based on normalizing transformations for non-Gaussian data [Text] / S. B. Prykhodko // Computational Intelligence (Results, Problems and Perspectives): Proceedings of the International Conference, May 12-15, 2015, Kyiv-Cherkasy, Ukraine / Ministry of Education and Science of Ukraine, Taras Shevchenko National University of Kyiv and [etc]; Vitaliy Ye. Snytyuk (Editor). – Cherkasy: editor July Chabanenko, 2015. – p.286-287. – ISBN 978-966-493-975-8

9. Кендалл, М. Теория распределений: Пер. с англ. [Текст] / М. Кендалл, А. Стюарт. Под ред. А. Н. Колмогорова. – М.: Наука. Гл. ред. физ.-мат. лит., 1966. – 588 с.

10. Приходько С.Б. Методичні вказівки до виконання лабораторних робіт з дисципліни “Обробка експериментальних даних на ЕОМ” [Текст] / С. Б. Приходько. – Миколаїв: НУК, 2005. – 52 с.

11. Коваленко, І.І. Сучасні методи статистичного аналізу даних [Текст] / І. І. Коваленко, С. Б. Приходько, Л. О. Латанська. – Николаев: НУК, 2011. – 185 с. – ISBN 978-966-321-118-3.

12. Приходько, С.Б. Аналитическая зависимость для выбора распределения Джонсона семейства S_L [Текст] / С. Б. Приходько, Л. Н. Макарова // Вестник ХНТУ. – Херсон: ХНТУ, 2012. – № 2 (45). – С.101–104.

ДОДАТОК А – Технічне завдання на розробку програми для моделювання тривалості створення застосунків на Java для ПК

Вступ

Назва розробки – “Програма для моделювання тривалості створення застосунків на Java для ПК”, далі програма.

Підстави для розробки

Завдання на дипломну роботу.

Призначення розробки:

Програма призначена для моделювання тривалості створення застосунків на Java для ПК. за 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 (значення випадкових величин).
- 2) Здійснювало вибір нормалізуючого перетворення Джонсона.
- 3) Виконувало нормалізацію емпіричних даних за вибраним перетворенням Джонсона.
- 4) Знаходило значення коефіцієнтів лінійного рівняння регресії за нормалізованими емпіричними даними.
- 5) Здійснювало побудову нелінійного рівняння регресії.
- 6) Обчислювало значення коефіцієнта детермінації R^2 .
- 7) Здійснювати побудову інтервалу прогнозування нелінійної регресії.
- 8) Виводити на екран нелінійну регресію, інтервал прогнозування нелінійної регресії та результати для моделювання тривалості (у місяцях)

створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

9) Виводити у текстовий файл результати для моделювання тривалості (у місяцях) створення застосунків на Java для ПК в залежності від трудомісткості (у людино-годинах).

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 транспортуватись 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	28.03.2022
2. Ескізний проект	Розробка ескізного проекту	24.04.2022
	Затвердження ескізного проекту	26.04.2022
3. Технічний проект	Розробка технічного проекту	08.05.2022
	Затвердження технічного проекту	10.05.2022
4. Робочий проект	Розробка програми	05.06.2022
	Розробка програмної документації	08.06.2022
	Випробування програми	10.06.2022

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

```

//Моделювання тривалості створення застосунків на Java для ПК
f=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f,'position',[0,0,700,300])
//Устанавливаем заголовок окна.
set(f,'figure_name',...
'Моделювання тривалості створення застосунків на Java для ПК');

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

lab_text0=uicontrol(f,'style','text','string',...
'Файл з даними','position',...
[360,150,100,20]);
edit_text=uicontrol(f,'style','edit','string',...
'D:\Data_F\Dur_PC_Java_34.txt','position',[350,120,100,20]);

BtDatInp=uicontrol('style','pushbutton','string','Ввести',...
'CallBack','[x,y,f1,n]=DataInput()','position',[500,120,100,20]);

lab_text1=uicontrol(f,'style','text','string',...
'Параметри нормалізуючого перетворення Джонсона','position',...
[100,270,300,20]);
lab_text2=uicontrol(f,'style','text','string',...
'Вибір типу перетворення','position',...
[500,270,200,20]);
lab_text3=uicontrol(f,'style','text','string',...
'Дані для моделювання','position',...
[80,150,120,20]);
lab_effort=uicontrol(f,'style','text','string',...
'Трудомісткість (у людино-годинах) =','position',...
[44,120,205,20]);
edit_effort=uicontrol(f,'style','edit','string',...
'500','position',[250,120,60,20]);
lab_k=uicontrol(f,'style','text','string',...
'Кількість значень, що моделюється =','position',...
[44,100,205,20]);
edit_k=uicontrol(f,'style','edit','string',...
'1','position',[250,100,60,20]);
//Создание текстовых полей для подписей полей ввода
//коэффициентов.
//Для X
//Подпись A=.
lab_ax=uicontrol(f,'style','text','string','Gamma_X=','position',...
[50,250,100,20]);
//Подпись B=.
lab_bx=uicontrol(f,'style','text','string','Nu_X=','position',...
[150,250,100,20]);
//Подпись C=.
lab_cx=uicontrol(f,'style','text','string','Fi_X=','position',...
[250,250,100,20]);
//Подпись D=.
lab_dx=uicontrol(f,'style','text','string','Lambda_X=','position',...
[350,250,100,20]);
//Для Y

```

```

//Подпись A=.
lab_ay=uicontrol(f,'style','text','string','Gamma_Y','position',...
[50,200,100,20]);
//Подпись B=.
lab_by=uicontrol(f,'style','text','string','Nu_Y','position',...
[150,200,100,20]);
//Подпись C=.
lab_cy=uicontrol(f,'style','text','string','Fi_Y','position',...
[250,200,100,20]);
//Подпись D=.
lab_dy=uicontrol(f,'style','text','string','Lambda_Y','position',...
[350,200,100,20]);
//Поле редактирования для ввода коэффициента a.
edit_ax=uicontrol(f,'style','edit','string','4.3573688','position',...
[50,230,100,20]);
//Поле редактирования для ввода коэффициента b.
edit_bx=uicontrol(f,'style','edit','string','0.5912588','position',...
[150,230,100,20]);
//Поле редактирования для ввода коэффициента c.
edit_cx=uicontrol(f,'style','edit','string','67.50','position',...
[250,230,100,20]);
//Поле редактирования для ввода коэффициента d.
edit_dx=uicontrol(f,'style','edit','string','1342643.20','position',...
.
[350,230,100,20]);
//
edit_ay=uicontrol(f,'style','edit','string','7.20870','position',...
[50,180,100,20]);
//Поле редактирования для ввода коэффициента b.
edit_by=uicontrol(f,'style','edit','string','1.5329788','position',...
[150,180,100,20]);
//Поле редактирования для ввода коэффициента c.
edit_cy=uicontrol(f,'style','edit','string','0.7037417','position',...
[250,180,100,20]);
//Поле редактирования для ввода коэффициента d.
edit_dy=uicontrol(f,'style','edit','string','301.669344','position',...
.
[350,180,100,20]);
//
//Текстовое поле, определяющее вывод результатов.
textresult=uicontrol(f,'style','text','string','', 'position',...
[5,80,650,20]);
//Флажок, отвечающий за выбор типа преобразования.
radio_bikv_x=uicontrol('style','radiobutton','string',...
'SU (включити), SB (виключити)', 'value',1,'position',...
[500,230,300,20]);
//
radio_bikv_y=uicontrol('style','radiobutton','string',...
'SU (включити), SB (виключити)', 'value',1,'position',...
[500,180,300,20]);
//

Btnr_pred_int=uicontrol('style','pushbutton','string','Змодельувати','C
allBack',...
' [yr,piyr_min,piyr_max,r_2,xi,yi,xs,ys]=Software_Dur_Sim(x,y,n) ',...
'position',[50,50,100,20]);

```

```

BtOutput=uicontrol('style','pushbutton','string','Побудувати','CallBac
k',...
'[f2]=Output_yr(xs,ys,xi,yr,piyr_min,piyr_max)','position',[175,50,100
,20]);

BtClose=uicontrol('style','pushbutton','string','Закрити',...
'CallBack','_Close()','position',[300,50,100,20]);

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

file_PC=mopen(dir_name,'r');
file_PC2=mopen('D:\PC_data2.txt','w+');
i=1;
//n=52;
while (feof(file_PC))<1
//for i=1:n
    x(i)=mfscanf(file_PC,'%f'); y(i)=mfscanf(file_PC,'%f');
    i=i+1;
end;
n=i-1;
for i=1:n
    fprintf(file_PC2,'%f\t%f\n',x,y);
    // fprintf(file_PC2,'%d\n',x);
end;
fclose(file_PC);
fclose(file_PC2);
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 міс.) створення
застосунків на Java для ПК в залежності від трудомісткості у люд.-
год.');
```

plot2d(x,y,-9);

endfunction

//Функция моделирования тривалості створення застосунків на Java для ПК

```

function [yr, piyr_min, piyr_max, r_2, xi, yi, xs,
ys]=Software Dur Sim(x, y, n)
//Считываем значение переменных из текстовых полей и
//преобразовываем их к числовому типу.
ax=eval(get(edit_ax,'string'));
bx=eval(get(edit_bx,'string'));
cx=eval(get(edit_cx,'string'));
dx=eval(get(edit_dx,'string'));
ay=eval(get(edit_ay,'string'));
by=eval(get(edit_by,'string'));
cy=eval(get(edit_cy,'string'));
dy=eval(get(edit_dy,'string'));

//Проверяем значение флажка, если флажок выключен,
if (get(radio_bikv_x,'value')==0)&(get(radio_bikv_y,'value')==0)
    zxA=0;
    for i=1:n
        zx(i)=ax+bx*log((x(i)-cx)/(dx-x(i)+cx));
        zy(i)=ay+by*log((y(i)-cy)/(dy-y(i)+cy));
    end
end

```

```

        zxA=zxA+zx(i);
    end;
    zxA=zxA/n;
    S2zx=0;
    for i=1:n
        S2zx=S2zx+(zx(i)-zxA)^2;
    end;

    b0=0; b1=0.9424;
    // tn=2.01;

    // Df=n-1;
    Df=n-2;
    P=0.975;
    Q=1-P;
    // Df=n-2;
    // Q=0.05/2/n;
    // P=1-Q;

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

    // Delta_z=T/n^0.5;
    Delta_z=T*(1+1/n)^0.5;

    for i=1:n
        zr(i)=b1*zx(i);
        yr(i)=(cy+dy)*exp((zr(i)-ay)/by)+cy/(1+exp((zr(i)-ay)/by));

        Delta_z=T*(1+1/n+(zx(i)-zxA)^2/S2zx)^0.5;
        pizrmin=zr(i)-Delta_z;
        pizrmax=zr(i)+Delta_z;

        piyr_min(i)=(cy+dy)*exp((pizrmin-ay)/by)+cy/(1+exp((pizrmin-ay)/by));
        piyr_max(i)=(cy+dy)*exp((pizrmax-ay)/by)+cy/(1+exp((pizrmax-ay)/by));

    end;

    //set(textresult,'string','Нелінійне рівняння регресії знайдено');
    y_mean=0;
    for i=1:n
        y_mean=y_mean+y(i);
    end;
    y_mean=y_mean/n;
    sy_t=0;
    sy=0;
    for i=1:n
        sy=sy+(y(i)-y_mean)^2;
        sy_t=sy_t+(y(i)-yr(i))^2;;
    end;
    r_2=1-sy_t/sy;
    set(textresult,'string',...
    sprintf("Моделювання виконано за нелінійним рівнянням регресії з \tR^2=%1.4f",r_2));
else
    set(textresult,'string',...

```

```

        'Нелінійне рівняння регресії не знайдено, моделювання не
        виконано');

end;

//
n_m=900;
Delta_z=T*(1+1/n)^0.5;

for i=1:n_m
    xi(i)=55+50*i;
    zxi=ax+bx*log((xi(i)-cx)/(dx-xi(i)+cx));
    zri=b1*zxi;
    yr(i)=(cy+dy)*exp((zri-ay)/by)+cy/(1+exp((zri-ay)/by));
    yi(i)=yr(i);

    Delta_z=T*(1+1/n+(zxi-zxA)^2/S2zx)^0.5;
    pizrmin=zri-Delta_z;
    pizrmax=zri+Delta_z;

    piyr_min(i)=(cy+dy)*exp((pizrmin-ay)/by)+cy/(1+exp((pizrmin-
ay)/by));
    piyr_max(i)=(cy+dy)*exp((pizrmax-ay)/by)+cy/(1+exp((pizrmax-
ay)/by));

end;

effort=eval(get(edit_effort,'string'));
k=eval(get(edit_k,'string'));
// k=3;
r = grand(k, 1, "unf", -1, 1);
[xtan]=tan(0.57*r);
[zd]=0.532275*asinh(3.874868*xtan);
for i=1:k
    xs(i)=effort;
    zxi=ax+bx*log((xs(i)-cx)/(dx-xs(i)+cx));
    zri=b1*zxi+zd(i);
    ys(i)=(cy+dy)*exp((zri-ay)/by)+cy/(1+exp((zri-ay)/by)),
end;

endfunction

function Outputy(x, y)
f1=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f1,'position',[50,50,700,300]);
//Устанавливаем заголовок окна.
set(f1,'figure_name','Емпіричні дані');
plot2d(x,y,-9);
endfunction

function [f2]=Output_yr(xs, ys, xi, yr, piyr_min, piyr_max)
f2=figure();//Создание графического окна.
//Устанавливаем размеры окна.
set(f2,'position',[100,100,700,300]);
//Устанавливаем заголовок );окна.
set(f2,'figure_name',...

```

```

'Моделювання тривалості (у міс.) створення застосунку на Java для ПК
за трудомісткістю (у люд.-год.)');
plot2d(xi,piyr_max,5);
plot2d(xs,ys,-9);
plot2d(xi,yr,1);
plot2d(xi,piyr_min,5);
endfunction

//Функция закрытия окна.
function _Close()
close(f);
close(f1);
close(f2);
endfunction

```

ДОДАТОК В – Опис програми

Загальні відомості

Програма для моделювання тривалості створення застосунків на Java для ПК для системи моделювання Scilab 6.0.0. Файл програми SoftwareDurSim.sce має розмір 11 КБ.

Функціональне призначення

Програма SoftwareDurSim призначений для моделювання тривалості створення застосунків на Java 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

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

ДОДАТОК Г – Інструкція користувача

Загальний опис роботи з ПЗ

Після завантаження файлу SoftwareDurSim.sce на екрані з'являється форма для відображення початкового стану програми для моделювання тривалості створення застосунків на Java для ПК (рисунок Г.1).

Моделювання тривалості створення застосунків на Java для ПК

File Tools Edit ?

Моделювання тривалості створення застосунків на Java для ПК

Параметри нормалізуючого перетворення Джонсона

Gamma_X=	Nu_X=	Fi_X=	Lambda_X=
4.3573688	0.5912588	67.50	1342643.20
Gamma_Y=	Nu_Y=	Fi_Y=	Lambda_Y=
7.20870	1.5329788	0.7037417	301.669344

Вибір типу перетворення

☒ SU (включити), SB (виключити)

☒ SU (включити), SB (виключити)

Дані для моделювання

Трудомісткість (у людино-годинах) = 500

Кількість значень, що моделюється = 1

Файл з даними

ur_PC_Java_34.txt

Ввести

Змоделюв... Побудувати Закрити

Рисунок Г.1 – Форма для відображення початкового стану програми

Далі у вікні редагування «Файл з даними» графічного вікна «Моделювання тривалості створення застосунків на Java для ПК» (рис.Г.1) потрібно відредагувати ім'я текстового файлу з початковими емпіричними даними, що будуть використовуватися для моделювання тривалості створення застосунків на Java для ПК.

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

Потім у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» (рис. Г.1) потрібно за допомогою двох радіо кнопок здійснити вибір типу (сім'ї) перетворення Джонсона і відредагувати його параметри у відповідних вікнах редагування «Параметри нормалізуючого перетворення Джонсона». Також у відповідних вікнах потрібно відредагувати трудомісткість (у людино-годинах) та кількість значень, що моделюються. Далі потрібно натиснути на командну кнопку «Змоделювати». Якщо вибір типу (сім'ї) перетворення Джонсона і редагування його параметрів зроблено хибно, то у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» з'являється текст «Нелінійне рівняння регресії не знайдено, моделювання не виконано».

Після цього знову у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» головного модуля програми (рис. Г.1) потрібно за допомогою двох радіо кнопок здійснити правильний вибір типу (сім'ї) перетворення Джонсона і коректне редагування його параметрів у відповідних вікнах редагування «Параметри нормалізуючого перетворення Джонсона». Далі натиснути на командну кнопку «Змоделювати». Якщо все було зроблено правильно, то у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» з'являється текст «Моделювання виконано за нелінійним рівнянням регресії» і виводиться значення коефіцієнту детермінації.

Потім у графічному вікні «Моделювання тривалості створення застосунків на Java для ПК» головного модуля програми (рис. Г.1) потрібно натиснути на командну кнопку «Побудувати». Це приведе до появи графічного вікна «Моделювання тривалості (у міс.) створення застосунків на Java для ПК за трудомісткістю (у люд.-год.)».

І на останок у графічному вікні «Моделювання тривалості створення застосунків на Java 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 моделювання тривалості створення застосунків на Java для ПК SoftwareDurSim.

Ціль випробувань: перевірка відповідності програми SoftwareDurSim 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 SoftwareDurSim. 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.5).

Моделювання тривалості створення застосунків на Java для ПК

File Tools Edit ?

Моделювання тривалості створення застосунків на Java для ПК

Параметри нормалізуючого перетворення Джонсона

Gamma_X=	Nu_X=	Fi_X=	Lambda_X=
4.3573688	0.5912588	67.50	1342643.20
Gamma_Y=	Nu_Y=	Fi_Y=	Lambda_Y=
7.20870	1.5329788	0.7037417	301.669344

Вибір типу перетворення

☒ SU (включити), SB (виключити)

☒ SU (включити), SB (виключити)

Дані для моделювання

Трудомісткість (у людино-годинах) = 500

Кількість значень, що моделюється = 1

Файл з даними

ur_PC_Java_34.txt

Ввести

Змоделюв... Побудувати Закрити

Рисунок Д.1 – Графічне вікно головного модуля програми

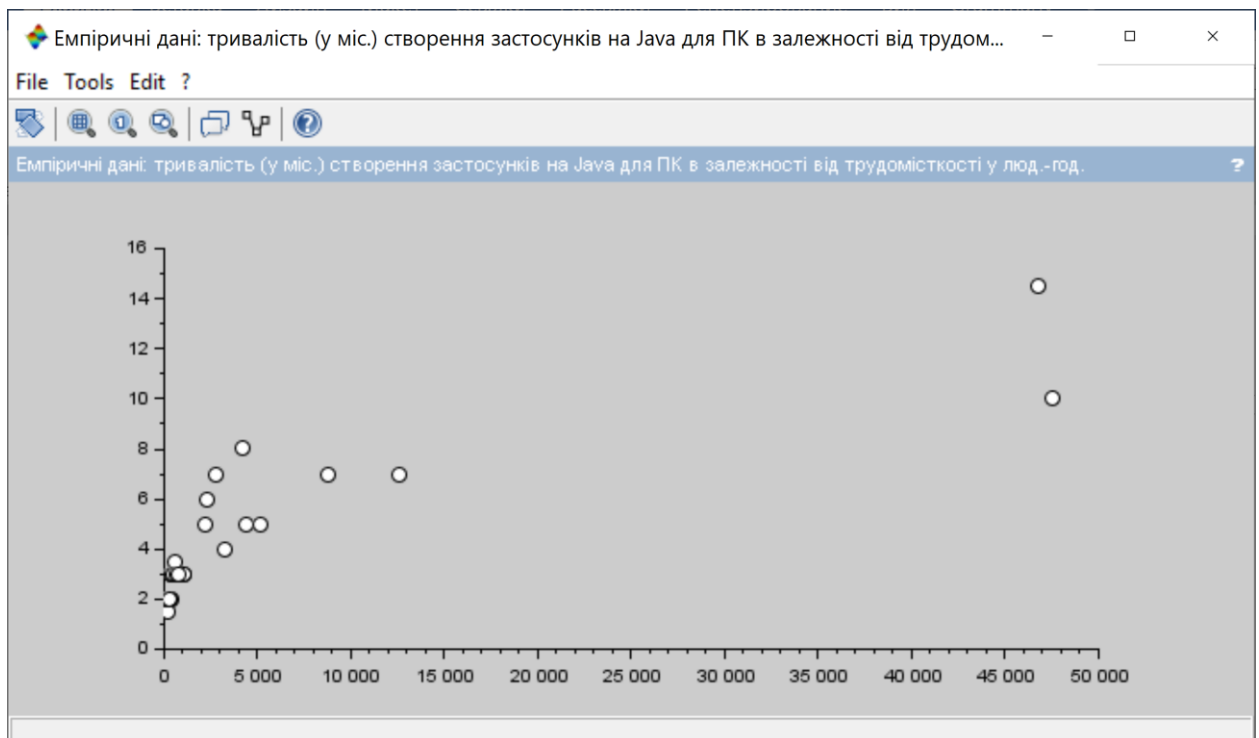


Рисунок Д.2 – Візуалізація тестових емпіричних даних

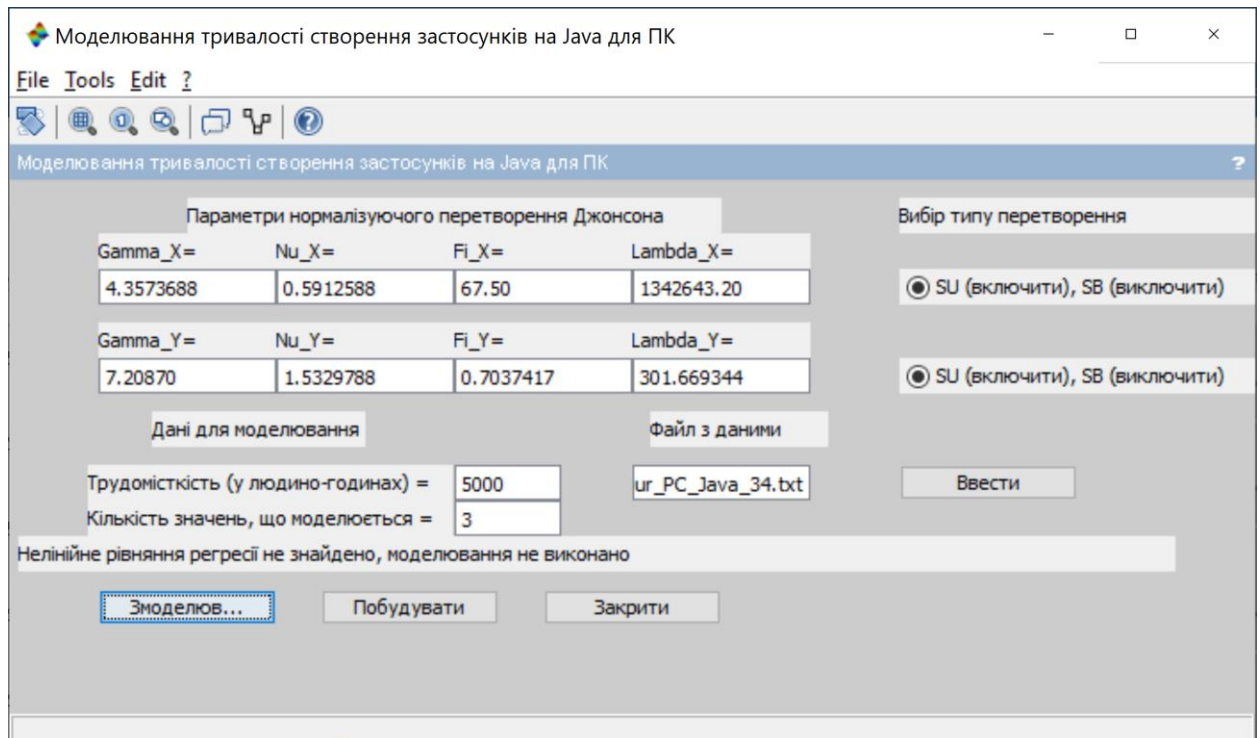


Рисунок Д.3 – Результати роботи програми після помилкового вибору типу (сім'ї) перетворення Джонсона і редагування його параметрів

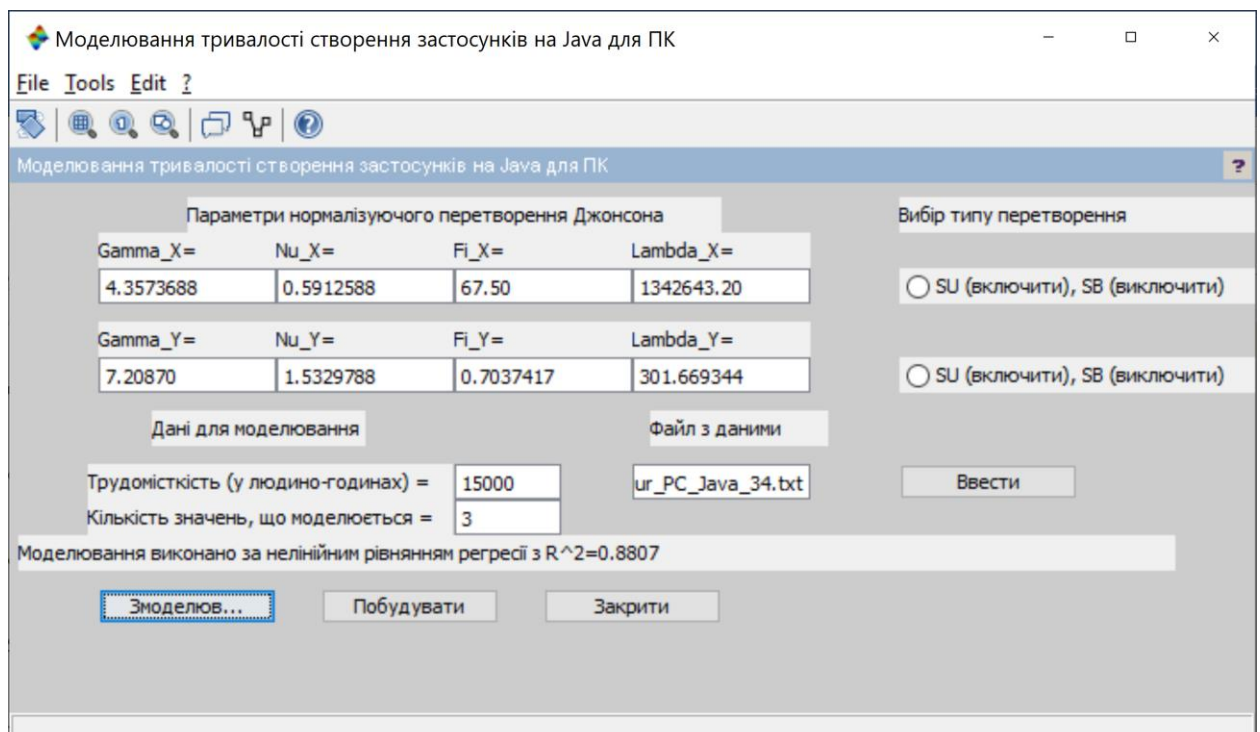


Рисунок Д.4 – Результати роботи програми після правильного вибору типу перетворення Джонсона і редагування його параметрів

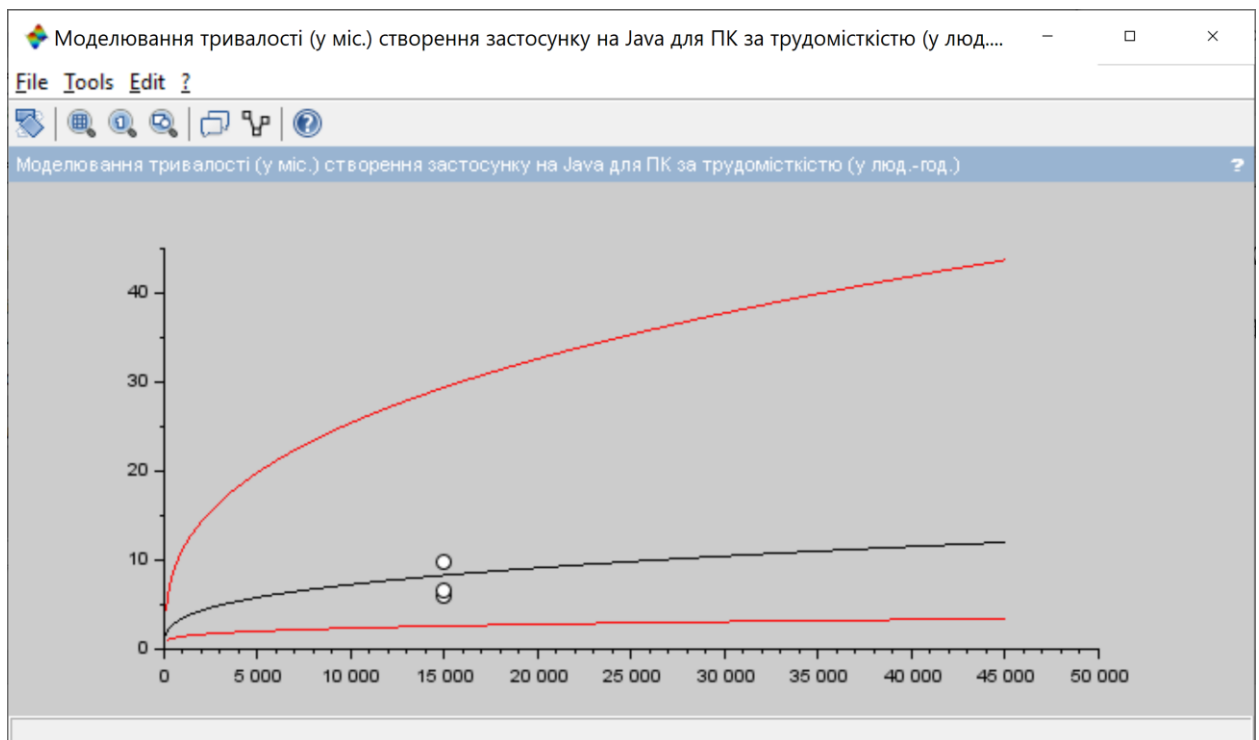


Рисунок Д.5 – Графічне вікно «Моделювання тривалості (у міс.) створення застосунків на Java для ПК за трудомісткістю (у люд.-год.)»

При випробуванні програми потрібно ввести наступні тестові дані із текстового файлу:

```

4372 5
385 2
2786 7
429 3
408 3
46787 14.5
2330 6
210 2
694 3
355 3
535 3.5
489 3
5175 5

```

12563 7
245 2
491 3
8772 7
600 3
3213 4
606 3
175 2
1018 3
509 3
167 1.5
899 3
105 2
47493 10
4224 8
147 2
669 3
2158 5
1007 3
784 3
245 2