

**Наукова робота на конкурс за напрямом:
Цивільна безпека (Охорона праці)**

на тему:

**ПРОГРАМНА ОЦІНКА ФУНКЦІОНАЛЬНОГО СТАНУ
ОПЕРАТОРІВ АВТОМАТИЗОВАНИХ ВИРОБНИЦТВ**

2020

ЗМІСТ

ВСТУП.....	3
1 ПРОБЛЕМА ВИЗНАЧЕННЯ ФУНКЦІОНАЛЬНОГО СТАНУ ОПЕРАТОРІВ АВТОМАТИЗОВАНИХ ВИРОБНИЦТВ	5
1.1 Автоматизоване виробництво як людино-машинна система	5
1.2. Задача оцінки функціонального стану людей-операторів в системах ергономічного забезпечення автоматизованих виробництв	7
1.3 Висновки до розділу 1	8
2 АНАЛІЗ МЕТОДІВ ВИРІШЕННЯ ЗАДАЧ ВИЗНАЧЕННЯ ФУНКЦІОНАЛЬНОГО СТАНУ ОПЕРАТОРІВ.....	9
2.1 Аналіз біометричних методів	9
2.2 Методи дослідження оцінки функціонального стану оператора	13
2.3 Висновки до розділу 2.....	17
3 РОЗРОБКА МАТЕМАТИЧНОЇ МОДЕЛІ ТА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОЦІНКИ ФУНКЦІОНАЛЬНОГО СТАНУ ОПЕРАТОРА.....	18
3.1 Постановка задачі	18
3.2 Математична модель для визначення функціонального стану людини- оператора	19
3.3 Результат реалізації інформаційної технології.....	22
3.4 Висновки до розділу 3.....	27
ВИСНОВКИ.....	28
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	29
ДОДАТОК А – Лістинг програмного коду.....	31
ДОДАТОК Б – Інструкція користувача	63
ДОДАТОК В – Експериментальні дослідження	76

ВСТУП

Актуальність. Висока економічна ефективність, технологічна цілісність, а також і експлуатаційна необхідність несе великий вплив на поширення автоматизації на виробництвах. Застосування сучасних технологій забезпечують стабільну якість виконання поставлених завдань за рахунок адаптованості операторів до їх робочих місць. Самопочуття людей завдає неабиякого впливу на цілісну роботу виробництв, а також ставить відбиток на їх здоров'ї, що призводить в подальшому до недієздатності. «Людський фактор» неодноразово досліджували, але задача автоматизованого визначення функціонального стану людей-операторів автоматизованих виробництв залишилася на етапі удосконалення досліджень.

Об'єкт дослідження. Діяльність операторів автоматизованих виробництв.

Предмет дослідження. Програмна оцінка функціонального стану операторів .

Мета роботи. Розробити програму для оцінки функціонального стану операторів автоматизованих виробництв.

Наукова новизна. На відміну від існуючих апаратних методів запропоновано невибагливий до технічного забезпечення метод, орієнтований на аналіз клавіатурного почерку з використанням нейронних мереж Кохонена.

Публікації. Оpubліковано 6 наукових робіт.

Апробація. Результати роботи доповідались на конференціях:

- Міжнародна науково-технічна конференція «Інформатика, Математика, Автоматика ІМА::2018»(Суми, 05-09 лютий 2018);
- XII Міжнародна науково-практична конференція «Цифрові технології в освіті, науці, суспільстві» (Петрозаводск, 4-6 грудня 2018 года);
- X Міжнародна студентська конференція «Перший крок у науку»;
- Міжнародна науково-технічна конференція «Інформатика, Математика, Автоматика ІМА::2019»(Суми, 23-26 лютий 2019 року);

- II тур Всеукраїнського конкурсу студентських наукових робіт з Комп'ютерної інженерії (Хмельницький, 27-28 березня 2019 року);
- Всеукраїнський конкурс студентських наукових робіт з напрямку «Інформатика і кібернетика» (Вінниця, 25-26 квітня 2019 року).

Впровадження. Результати впроваджені в навчальний Сумського державного університету.

1 ПРОБЛЕМА ВИЗНАЧЕННЯ ФУНКЦІОНАЛЬНОГО СТАНУ ОПЕРАТОРІВ АВТОМАТИЗОВАНИХ ВИРОБНИЦТВ

1.1 Автоматизоване виробництво як людино-машинна система

Традиційною задачею автоматизації є:

- підвищення виробництва;
- виключення ручної праці;
- виключення некваліфікованої праці.

Автоматизовані виробництва мають три головні ознаки, що виділяють їх серед інших підприємств [1].

По-перше, управління здійснюється технологічним об'єктом управління (ТОУ).

По-друге, управління здійснюється в реальних умовах з використанням масштабованих рамок часу, тобто в темпі протікання ТОУ.

По-третє, реалізація та вибір окремих задач для управління ТОУ з залученням автоматизованих технологічних засобів і людини-оператора, що працює з ними.

Об'єктом управління в автоматизованому виробництві виступають певні енергетичні та технологічні засоби або їх взаємопов'язані технологічні процеси. Ергатичним елементом у даному випадку виступає людина-оператор. Її роль полягає в дистанційному дослідженні, аналізу за роботою та виконанні функцій таких, як підключення, відключення, редагування між перемиканнями декількох існуючих режимів, керування впливовими факторами системи, оперування завантаженням необхідних даних, тощо.

Автоматизоване виробництво – різностороннє, тому має ряд типів за своїм призначенням [3]:

- інформаційний;
- інформаційний з рядом порад (комбінований);

- керований (цифрове керування).

Значну перевагу віддають інформаційному типу, адже за структурою автоматизоване виробництво (рис. 1.1) людина-оператор на пряму пов'язана з усіма складовими даного типу.

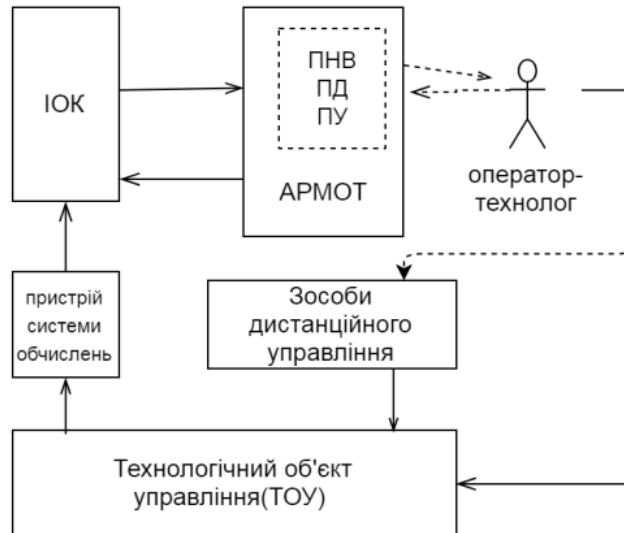


Рисунок 1.1 – Структура автоматизованого виробництва за інформаційним типом

Оператор може керувати ТОУ як власноруч, так і за допомогою системи дистанційного керування (за принципом "включено-виключено") [3,4].

Структура автоматизованого виробництва реалізує чотири рівня ієрархії функцій управління: ТОУ; виведення процесу на заданий режим; стабілізація заданого технологічного режиму; узагальнена інформація, рекомендації оператору.

Порівняно з системами ручного та дистанційного керування технологічними процесами дана структура має ряд переваг:

- обробка великого потоку інформації про хід технологічного процесу і про стан обладнання ТОУ;
- отримання оперативної техніко-економічної інформації про роботу автоматизованого виробництва;
- отримання інформації в концентрованому вигляді на одному пристрої наочного зображення у вигляді формуляра, графіка, словесних рекомендацій;

- документування ходу технологічного процесу і стану обладнання ТОУ в процесі експлуатації;
- отримання оператором оперативної інформації від наочного зображення у вигляді формуляра, графіка, словесних рекомендацій в складних виробничих ситуаціях [3].

1.2. Задача оцінки функціонального стану людей-операторів в системах ергономічного забезпечення автоматизованих виробництв

Сенс антропоцентричного підходу щодо дослідження автоматизованих систем полягає в тому, що кібернетичним засобом праці в автоматизованому виробництві виступає його технічна частина, яка є допоміжним елементом для покращення діяльності суб'єкта праці [1].

Мета ергономіки ("людського фактору") націлена на пізнанні людських здібностей та обмежень, а потім застосуванні цих навчань для покращення взаємодії людей з продуктами, системами та оточенням.

Ергономіка має на меті:

- покращити робочі місця та навколишнє середовище, щоб мінімізувати ризик травмування чи заподіяння шкоди;
- створити безпечні, комфортні та продуктивні робочі простори шляхом залучення людських здібностей та обмежень у дизайн робочого простору, включаючи розмір тіла, силу, вміння, швидкість, сенсорні здібності людини (зір, слух) та навіть ставлення.

За даними Safe Work Ukraine, загальна економічна вартість виробничих травм та захворювань оцінюється в 20 мільярдів доларів.

Дослідження показують, що незадовільний функціональний стан співробітників – це найпоширеніша у світі працездатність, яка впливає на працівників з офісів, пов'язаних з інформаційними системами, будівельних майданчиків та категорії найвищого ризику, сільського господарства.

Основні завдання ергономічного проектування [1]:

- визначення програми вимог;
- розподілення завдань (між технологіями та операторами), звертаючи увагу на потужність та обмеження;
- розгляд альтернатив попереднього пункту та вибір найкращого рішення;
- розробка завдань, з урахуванням організаційних аспектів (рівень освіти, командна робота, графік роботи, тощо) та з урахуванням фізичних особливостей людини (швидка/повільна втомлюваність);
- тестування у віртуальному середовищі за допомогою симуляції, моделей, тощо;
- проектування взаємодії інтерфейсів між людьми та АТК;
- адаптація конструкцій.

Кожна з вище зазначених задач має право на існування у випадку, якщо буде діяти ефективний та якісний контроль за функціональним станом операторів автоматизованих виробництв. Погане самопочуття несе ризик збоїв на підприємствах, в системах, але в першу чергу втрату людей-операторів на їх робочих місцях (пов'язане зі станом здоров'я: захворюваність як слабка форма (температура, тиск, запаморочення, тощо), так і складна (потрапляння до лікарні)).

Отже, першочерговою роллю та задачею оперативного управління автоматизованих виробництв є контроль за функціональним станом їх операторів.

1.3 Висновки до розділу 1

Аналіз автоматизованих виробництв виявив їх людино-машинний характер, напруженість діяльності операторів та проблему постійного контролю функціонального стану операторів.

Погане самопочуття несе ризик збоїв на підприємствах, в системах, але в першу чергу втрату або відсутність операторів на їх робочих місцях.

2 АНАЛІЗ МЕТОДІВ ВИРІШЕННЯ ЗАДАЧ ВИЗНАЧЕННЯ ФУНКЦІОНАЛЬНОГО СТАНУ ОПЕРАТОРІВ

2.1 Аналіз біометричних методів

Біометрика є найбільш доречним засобом для надійного та швидкого способу ідентифікації та автентифікації осіб за допомогою використання унікальних біологічних характеристик [6].

Біометрика дозволяє ідентифікувати та автентифікувати людину на основі сукупності впізнаваних і перевірених даних, унікальних та специфічних для них.

Біометрична автентифікація – це процес порівняння даних за характеристиками людини з біометричним «шаблоном» цієї людини для визначення схожості. Еталонна модель спочатку зберігається в базі даних або захищеному портативному елементі, наприклад смарт-картці, потім збережені дані порівнюються з біометричними даними людини, що підлягають автентифікації.

Біометрична ідентифікація полягає у визначенні особи людини. Метою є отримання біометричних даних у цієї людини. Це може бути фотографія її обличчя, запис її голосу або зображення її відбитка пальця. Потім ці дані порівнюються з біометричними даними кількох інших осіб, що зберігаються в базі даних.

Існує три можливі способи довести свою особистість:

- використання навички, яким володіє людина, за допомогою ключа автомобіля, документа, картки чи значка;
- використання особистих знань (ім'я, секрет або пароль);
- використання свого біометричного матеріалу (відбиток пальця, руку, обличчя, тощо).

Виходячи з перерахованих способів, переважає використання біометрики, адже провідним є рівень безпеки та точності, який вона гарантує.

На відміну від паролів, значків чи документів, біометричні дані не можна забувати, обмінювати, красти або підробляти.

За підрахунками, зробленими сером Френсісом Галтоном (двоюрідним братом Дарвіна), ймовірність знайти два подібних відбитка пальців одна з 64 мільярдів навіть з однаковими близнюками (гомозиготами) [4].

На сьогодні існує сім видів біометрики [7]:

- хімічний;
- візуальний;
- просторовий;
- поведінковий;
- нюховий;
- жилистий;
- слуховий.

До хімічного виду належить ДНК-відповідність (ідентифікація особи за допомогою аналізу сегментів з ДНК).

До візуального виду належать:

- вухо (ідентифікація особи за допомогою форми вуха);
- очі – розпізнавання радужки (використання ознак, знайдених в радужці, для ідентифікації особи);
- розпізнавання обличчя (аналіз особливостей чи моделей обличчя для аутентифікації або розпізнавання особистості, більшість систем розпізнавання обличчя або використовують власні інтерфейси, або локальний аналіз функцій);
- розпізнавання відбитків пальців (використання хребтів і долин (дрібниць), знайдених на поверхневих кінчиках пальця людини для ідентифікації особи).

До візуального та просторового видів відносять:

- розпізнавання геометрії пальця (використання 3D геометрії пальця для визначення ідентичності);

- розпізнавання геометрії руки (використання геометричних особливостей руки, таких як довжина пальців і ширина руки для ідентифікації особи).

До поведінкових належать:

- хода (використання індивідуального стилю ходьби чи ходи для визначення особи);

- розпізнавання введення тексту (використання унікальних характеристик осіб, які друкують для встановлення особи).

До візуального та поведінкового видів відносять розпізнавання підписів. Аутентифікація фізичної особи шляхом аналізу стилю рукописного тексту, зокрема підпису. Існує два ключових типи цифрової рукописної автентифікації підписів, статична та динамічна. Статичний – найчастіше візуальне порівняння між одним відсканованим підписом та іншим відсканованим підписом, або відсканованим підписом проти чорнильного підпису. Доступна технологія для перевірки двох відсканованих підписів за допомогою алгоритмів просування. Динамічна стає все більш популярною, оскільки дані церемонії збираються разом із X, Y, T і P координатами сигнатора від пристрою підписання. Ці дані можуть бути використані в суді за допомогою цифрових інструментів судово-медичної експертизи та для створення біометричного шаблону, за допомогою якого динамічні підписи можуть бути автентифіковані або під час підписання, або після підписання, і як тригери в робочих процесах.

До нюхового виду належить лише запах (використання індивідуального запаху для визначення особи).

До жилкового виду належить лише розпізнавання вен (тип біометрії, який можна використовувати для ідентифікації осіб на основі моделей вен на пальці або долоні людини).

До слухового виду належать:

- голос – ідентифікація динаміка (Ідентифікація – це завдання визначення особи невідомого мовця. Ідентифікація динаміка - це відповідність 1: N (багато), де голос порівнюється з N шаблонами. Системи ідентифікації

динаміків також можуть бути запроваджені таємно без відома користувача, щоб ідентифікувати учасників дискусії під час обговорення, автоматизувати системи автоматичного оповіщення про зміни динаміка, перевірити, чи користувач уже зареєстрований у системі тощо. Наприклад, поліцейський порівнює порівняльний ескіз нападник на базу даних раніше задокументованих злочинців, щоб знайти найближчу відповідність. У криміналістичних програмах звичайно спочатку здійснити процес ідентифікації оратора, щоб створити список «найкращих збігів», а потім виконати ряд процесів перевірки для визначення переконливої відповідності);

- голос – перевірка / аутентифікація динаміка (використання голосу як методу визначення особи динаміка для контролю доступу. Якщо оратор стверджує, що має певну особу, і голос використовується для перевірки цієї заяви. Перевірка динаміка – це відповідність 1: 1, коли голос одного доповідача відповідає одному шаблону (також його називають "голосовим друком" або "голосовою моделлю"). Перевірка спікера зазвичай використовується як "воротар", щоб забезпечити доступ до захищеної системи (наприклад, телефонний банкінг). Ці системи працюють із знаннями користувача і зазвичай вимагають їх співпраці. Наприклад, пред'явлення паспорта людини на прикордонному контролі - це процес перевірки – агент порівнює обличчя людини з малюнком у документі).

Біометричні технології діляться по типу досліджуваної ознаки на статичні (засновані на невід'ємних унікальних властивостях) і динамічні (засновані на психофізіологічних особливостях).

Статичні методи біометричних технологій [7], вимагають для свого застосування наявність дорогої апаратури для знімання інформації про користувача, а, отже, мають високу вартість.

Значну перевагу серед методів біометричних технологій отримують динамічні методи, адже вони дозволяють змінювати біометричний еталон (створені персональні дані конкретної людини) не витрачаючи на це великі кошти.

Серед динамічних методів, які побудовані на психофізіологічній особливості людини, вирізняють такі види розпізнавання:

- рукописний почерк;
- клавіатурний почерк;
- голосові характеристики.

2.2 Методи дослідження оцінки функціонального стану оператора

Клавіатурний почерк – це набір динамічних характеристик роботи на клавіатурі [9].

Сучасні дослідження показують, що клавіатурний почерк користувача має стабільність [8], що дозволяє ідентифікувати користувача, що працює з клавіатурою з високим ступенем точності.

Розпізнавання натискань клавіш не вимагає додаткового обладнання, з яким можна читати, сканувати, переглядати, записувати чи іншим чином допитувати запитуючого користувача, оскільки кожен комп'ютер обладнаний клавіатурою [14].

Для автентифікації особи розпізнавання натискань клавіші покладається виключно на програмне забезпечення, яке може знаходитися на клієнтській або хост-системі. Щоб створити шаблон реєстрації, людина повинна кілька разів вводити своє ім'я користувача та пароль. Найкращі результати отримують, якщо зарахування відбувається протягом певного періоду часу, а не за один засідання: протягом певного часу окремі характеристики визначаються більш точно. Користувач має розпізнавати натискання клавіш, не вводячи жодних виправлень. Якщо допущені помилки натискання клавіші, система запропонує користувачеві почати спочатку.

Деякі відмітні характеристики, що вимірюються системами розпізнавання натискань клавіш, є:

- тривалість утримування кожної клавіші;
- тривалість між натисканнями клавіш;

- швидкість набору тексту;
- тенденції перемикатися між цифровою клавіатурою та номерами клавіатури;
- послідовності натискань клавіш, що беруть участь у капіталізації.

Кожна окрема характеристика вимірюється і зберігається як унікальний шаблон. Деякі системи мають автентифікацію лише при вході, тоді як інші продовжують стежити за користувачем протягом усього сеансу. Як і в інших біометричних даних, зразок натискання клавіші користувача порівнюється із збереженим шаблоном, і доступ надається, якщо представлений зразок відповідає шаблону за попередньо встановленими ймовірностями.

Алгоритми розпізнавання клавіатурного почерку поділяються на три групи. Одна з них – це група, яка дозволяє забезпечити більшу точність, алгоритмів, які аналізують почерк після введення текстового фрагменту або фрази.

Одним із методів покращення роботи алгоритму є створення та постійне редагування еталонних значень. Ця дія дозволить вхідним даним не застаріти та завжди відповідати поточному рівню швидкості друку людини-оператора.

Вважається, що методи, засновані на використанні нейронних мереж, забезпечують більш високу точність, але потребують великих обчислювальних потужностей.

Класичне навчання нейронних мереж ділиться на дві категорії (рис. 2.1):

- з вчителем (Supervised Learning) – розглядає задачі з вже вказаним методом їх рішення;
- без вчителя (Unsupervised Learning) – самоорганізоване.

Neural Network Toolbox включає два типи самоорганізованих неконтрольованих мереж з навчанням без вчителя:

- конкурентні шари;
- самоорганізовані карти (мапи) [11].



Рисунок 2.1 – Різновиди класичного навчання нейромереж

Ці два типи входять до нейронної мережі Кохонена (рис. 2.2). Різниця між ними лише в тому, що конкурентні шари Кохонена – мережа з нейронами, які неупорядковано, а самоорганізована мапа Кохонена – мережа з нейронами, які впорядковано.

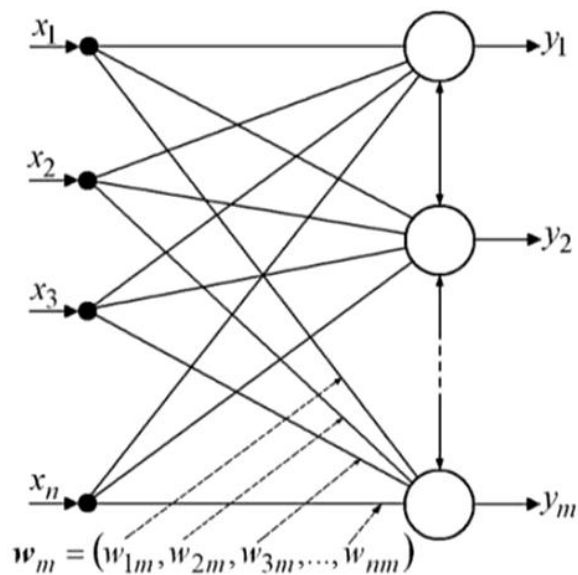


Рисунок 2.2 – Нейронна мережа Кохонена

Для формування мережі Кохонена необхідно вектор вхідних сигналів (2.1), вихідний вектор (2.2) та матриця множини вагових коефіцієнтів (2.3).

$$x(i) = (x_1(i), x_2(i), \dots, x_n(i)), \text{ де } i = 1, 2, \dots, P; \quad (2.1)$$

$$y = (y_1, y_2, \dots, y_m), \text{ де } m - \text{кількість кластерів}; \quad (2.2)$$

$$W = \begin{pmatrix} W^{11} & W^{12} & \dots & W^{1m} \\ W^{21} & W^{22} & \dots & W^{2m} \\ \vdots & \vdots & \dots & \vdots \\ W^{k1} & W^{k2} & \dots & W^{km} \end{pmatrix}, \quad (2.3)$$

де $W^{ij} = (w_1^{ij}, w_2^{ij}, \dots, w_n^{ij})$ – елементи матриці (вектори вагових коефіцієнтів).

Для вибору типу мережі для системи оцінки функціонального стану операторів інформаційних систем було визначено помилки функціонування кожної з них:

- Конкурентний шар Кохонена(news) – похибка = [3.104e-05];
- Самоорганізована мапа Кохонена(newsom) – похибка = [2.8276e-09].

Порівнюючи результати отриманих помилок отримано, що для визначення функціонального стану людини слід обрати Самоорганізовану мапу Кохонена, адже, порівнюючи її з іншим типом, помилка була нижчою.

Дану мережу застосовують не тільки для візуалізації результатів аналізу даних та кластеризації, а й для побудови структури багатовимірних даних [13].

Розпізнавання натискань клавіш є відносно новою біометричною технологією, достовірна інформація щодо його ефективності зближена до стовідсоткового результату.

Біометрика розпізнавання натискань, як правило, вважається найпростішою біометричною технологією для впровадження та використання. Жодне обладнання не бере участь. Програмне забезпечення може бути встановлено на клієнті або хості.

Оскільки аутентифікація заснована на звичайному введенні на клавіатурі, особам потрібно лише ввести передбачений текст для аутентифікації.

2.3 Висновки до розділу 2

Біометрика дозволяє ідентифікувати та аутентифікувати людину на основі сукупності пізнаваних і перевірених даних. На відміну від паролів, значків чи документів, біометричні дані не можна забувати, обмінювати, красти або підробляти.

Біометрика розпізнавання натискань, як правило, вважається найпростішою біометричною технологією для впровадження та використання.

Для реалізації розпізнавання за клавіатурним почерком вибрано алгоритм, який перевіряє імпульси, що поступають з клавіатури, порівнюючи їх з еталонним значенням, яке створила нейронна мережа.

Обрано невибагливу та максимально точну нейронну мережу Кохонена (різновид – Самоорганізована мережа Кохонена).

3 РОЗРОБКА МАТЕМАТИЧНОЇ МОДЕЛІ ТА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ОЦІНКИ ФУНКЦІОНАЛЬНОГО СТАНУ ОПЕРАТОРА

3.1 Постановка задачі

Розробити інформаційну технологію (рис. 3.1) для визначення функціонального стану оператора автоматизованих технологічних комплексах на їх робочих місцях.

Програма має вести перевірку за створеним еталонним вектором, попередньо кожним користувачем програми. Оператор за допомогою контрольних слів, введених з клавіатури, отримує результат його функціонального стану в даний момент часу та поради для вдосконалення самопочуття (за бажанням), а також має можливість переглянути попередні показники його тестування в системі визначення функціонального стану людини-оператора.

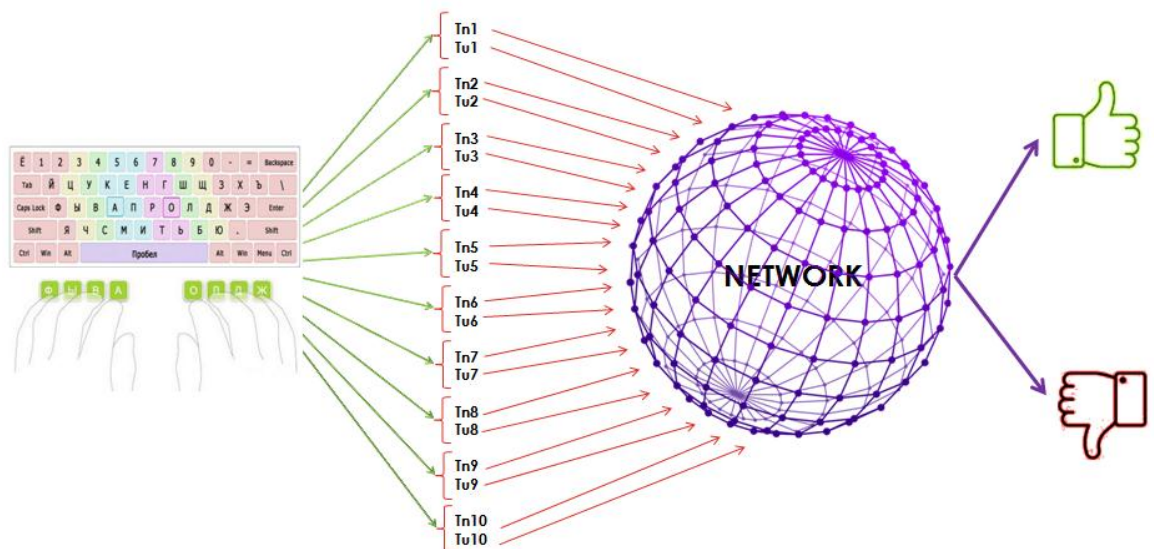


Рисунок 3.1 – Модель роботи системи

(tn_i – час натискання на клавішу, tu_i – час утримання клавіші)

3.2 Математична модель для визначення функціонального стану людини-оператора

Розроблення інформаційної системи оцінки функціонального стану людини-оператора включає в себе математичну модель з чотирьох основних етапів (рис. 3.2).

1. Формування навчальних даних – автоматичне створення всіх необхідних файлів, без втручання людини.
2. Введення контрольного слова – користувач вводить декілька разів контрольні слова.
3. Вирішення задачі кластеризації – використання нейронної мережі в середовищі MatLab для формування еталонного вектору даних, введених користувачем з клавіатури.

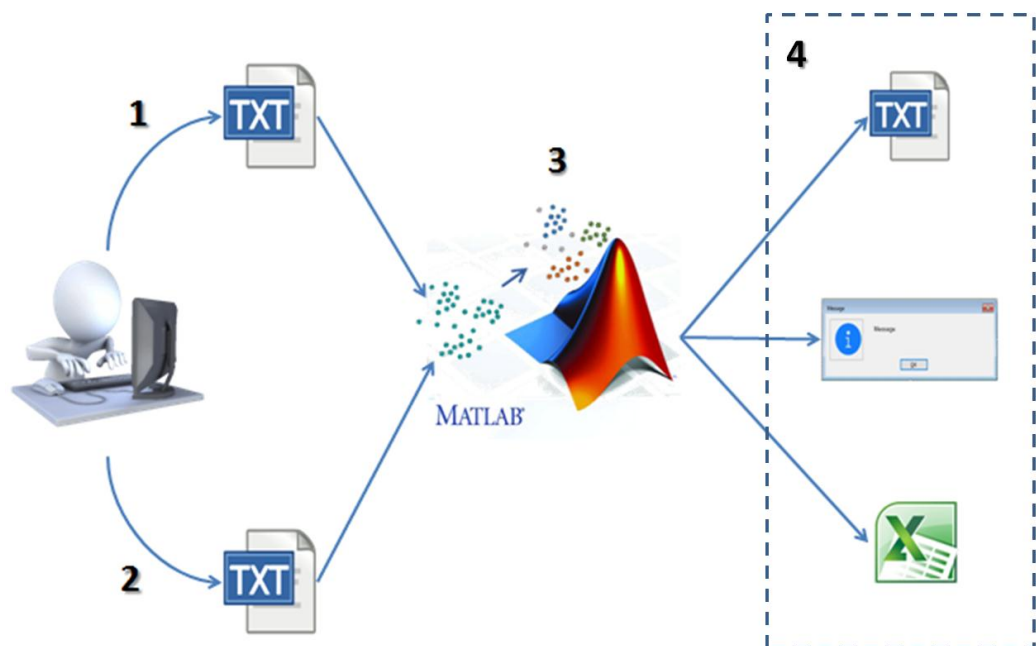


Рисунок 3.2 – Схема основної роботи програмного продукту

4. Результат визначення функціонального стану оператора (виведення в декілька етапів) (рис. 3.3).

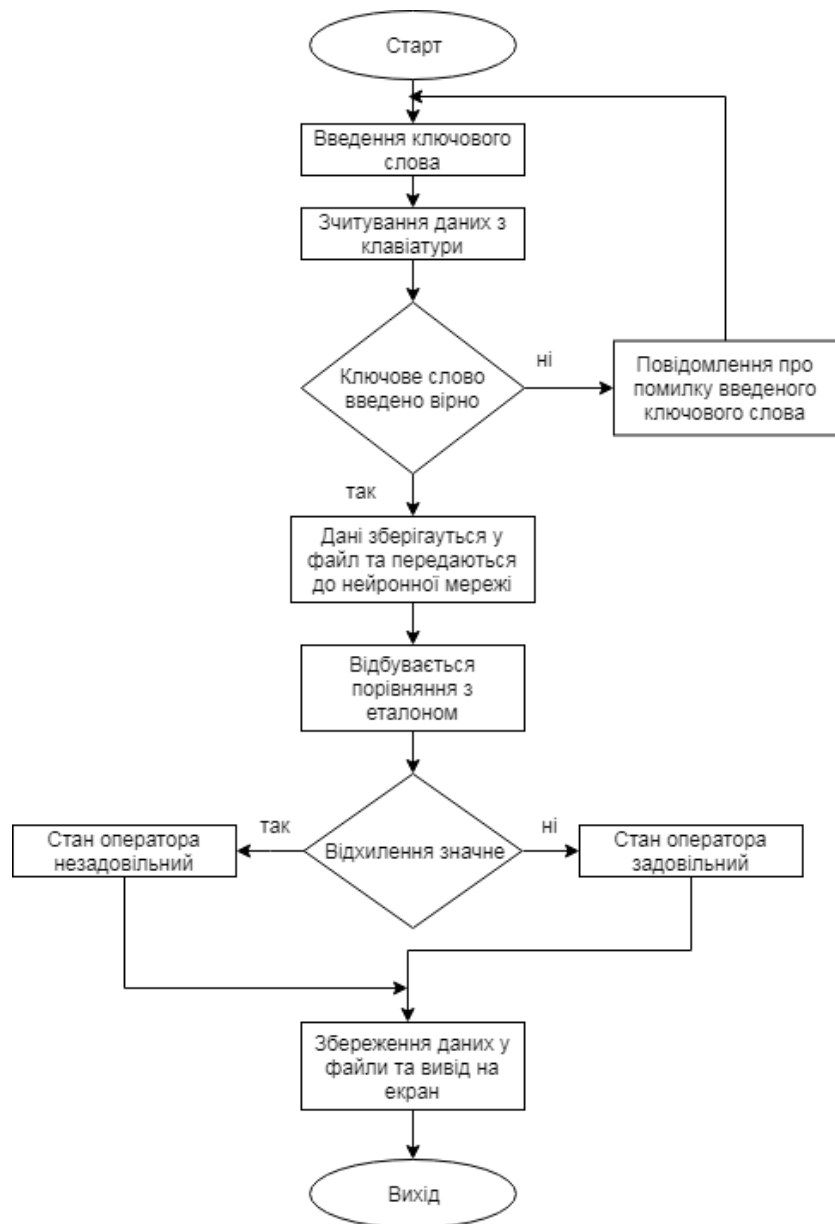


Рисунок 3.3 – Алгоритм визначення функціонального стану оператора
Вхідною інформацією математичної моделі виступають дані введені з
клавіатури.

$$\Delta t = t_{c-1} - t_c , \quad (3.1)$$

де Δt – час утримання клавіші або час між натисканнями клавіш;

t_{c-1} – тривалість попереднього утримання клавіші або між натисканням
клавіш;

t_c – час останнього утримання клавіші або час між натисканнями клавіш
(c – поточне значення).

Саме за формулою (3.1) відбувається запис зчитаних даних з клавіатури до файлів для подальшої їх передачі до нейронної мережі. Архітектура штучної нейронної мережі здатна обробити біометричні дані (рис. 3.4), вирішивши фільтрацію та кластеризацію надійшовши даних з клавіатури.

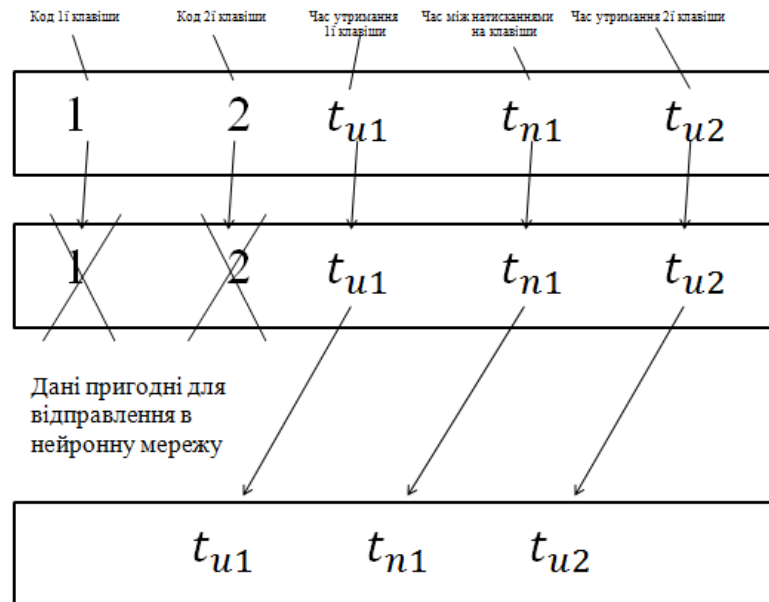


Рисунок 3.4 – Отримання нейронною мережею вхідних значень для створення навчальної вибірки

Алгоритм навчання Самоорганізованої мапи Кохонена має семи обов'язкових кроків для вирішення задачі кластеризації:

- визначити значення параметрів алгоритму;
- ініціалізація векторів навчальних ваг нейронів;
- перевірка на наявність елементів в навчальній виборці;
- розрахунок густини слою Кохонена для кожного вектору, що підлягає кластеризації; пошук нейрона-переможця; розрахунок та використання нових вагових векторів;
- збільшення ітерації на одну позицію;
- повернення до перевірки на наявність елементів в навчальній виборці.

3.3 Результат реалізації інформаційної технології

Встановлення та запуск інформаційної технології оцінки функціонального стану оператора (рис.3.5 – 3.6).

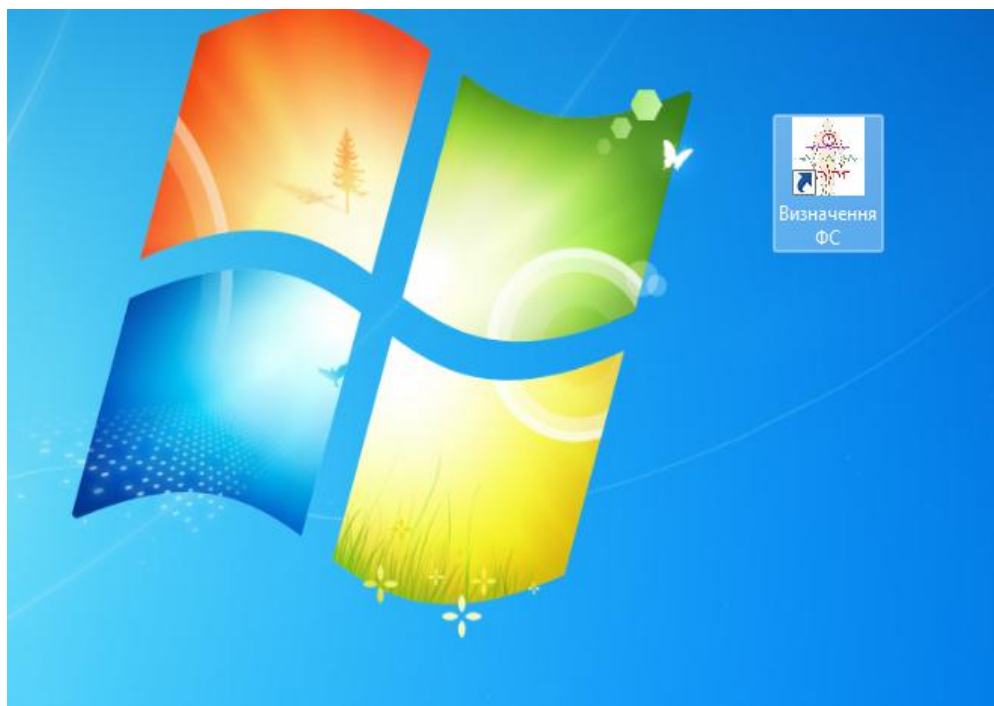


Рисунок 3.5 – Ярлик програми на Робочому столі

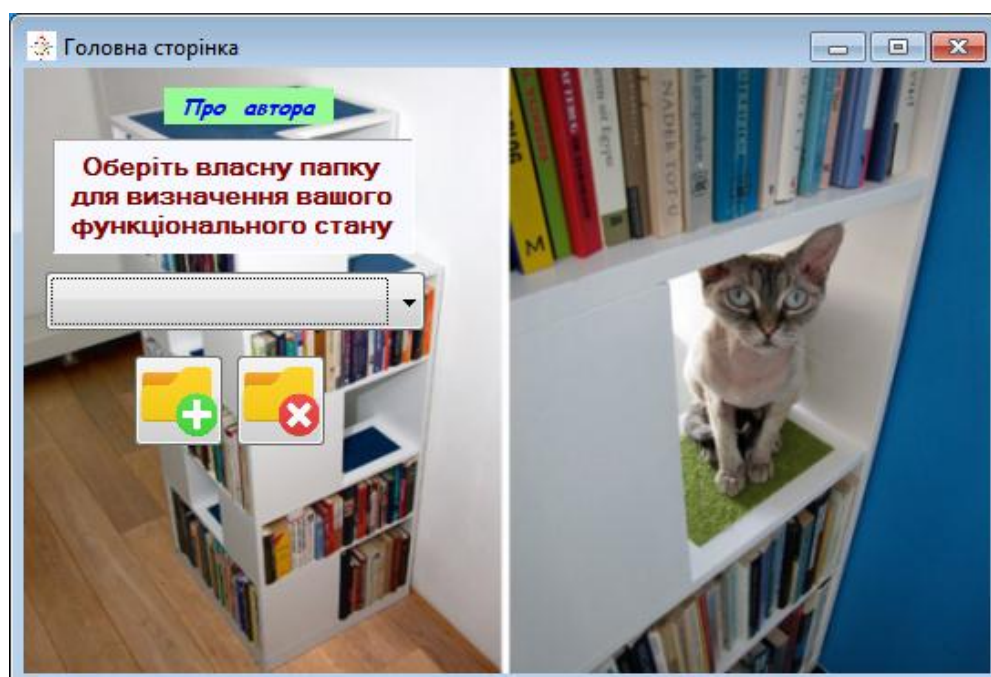


Рисунок 3.6 – Головне вікно програми

Натиснувши на одну з папок з випадного списку, користувачу відкривається вікно з вибором дій (рис. 3.7).

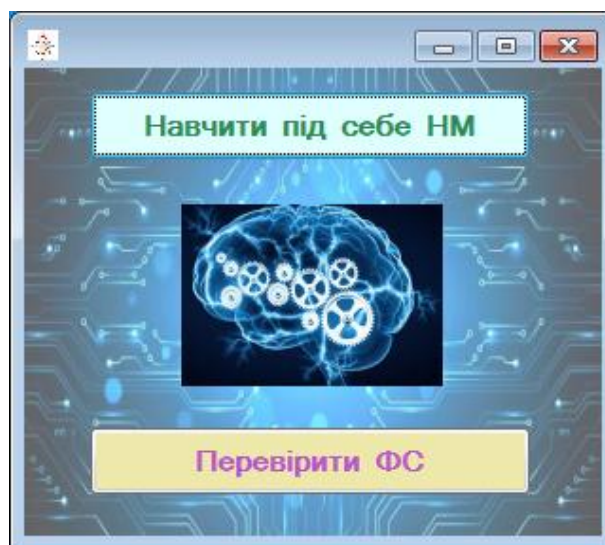


Рисунок 3.7 – Проміжне вікно програми

Обравши першу кнопку користувачу відкривається вікно «Персональне налаштування системи», де задаються вхідні значення до нейронної мережі, тобто створюються навчальні вибірки (рис. 3.8).

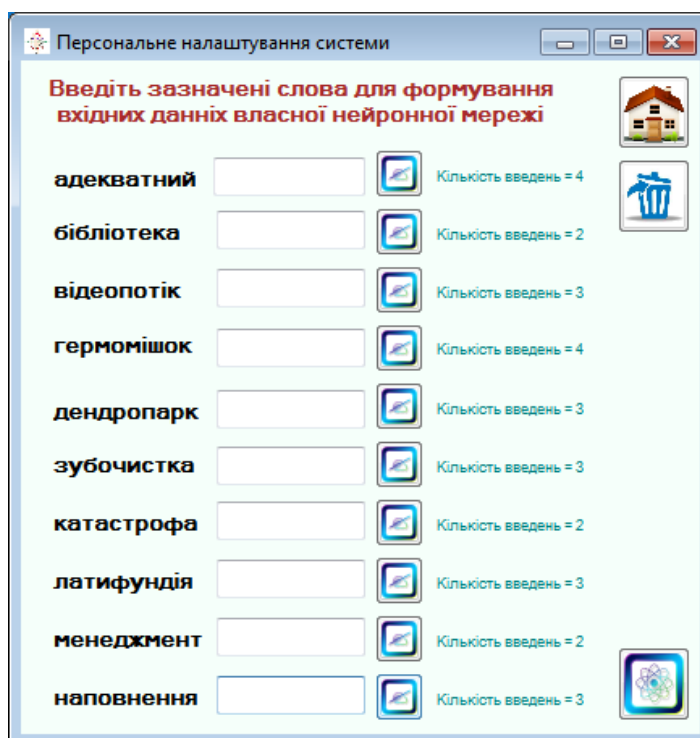


Рисунок 3.8 – Вікно програми «Персональне налаштування системи»

Всі збережені дані знаходяться в директорії диску C, створюються автоматично (рис. 3.9).

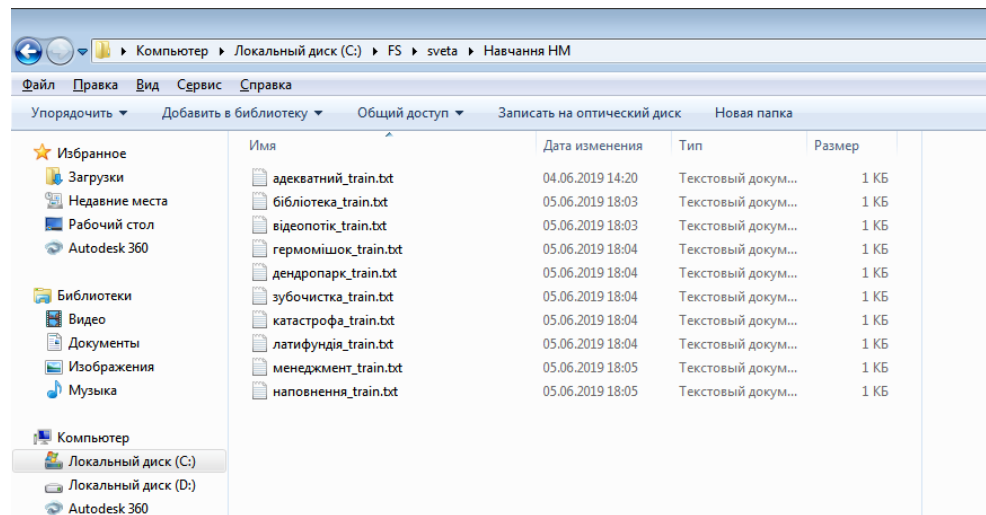


Рисунок 3.9 – Зображення всіх створених файлів

Після створення навчальної вибірки можна видалити за бажання файли, які їх містять, або ж перевірити власний функціональний стан, перейшовши до вікна «Перевірка поточного стану» (рис. 3.10).

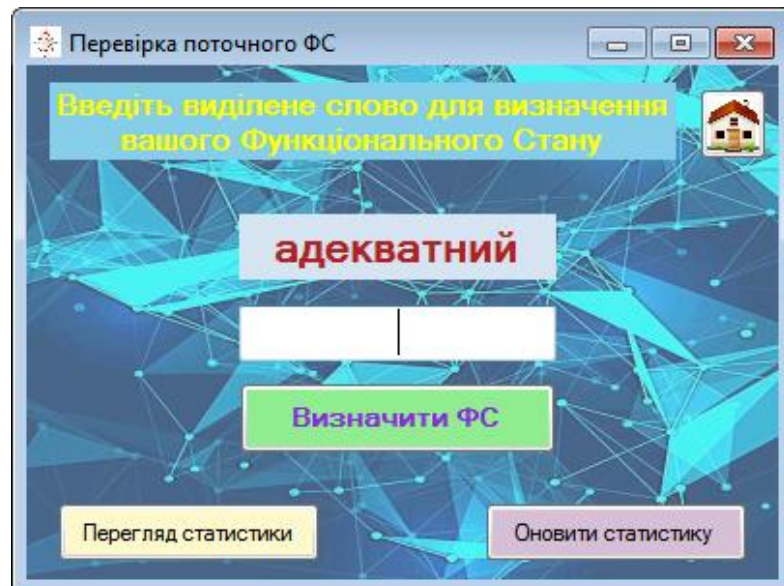


Рисунок 3.10 – Вікно програми, де перевіряється поточний стан користувача

Вводячи запропоноване слово та натискаючи на кнопку «Визначити ФС» йде перевірка оцінки функціонального стану за допомогою нейронної мережі та виводиться результат у вигляді інформаційного повідомлення (рис. 3.11).

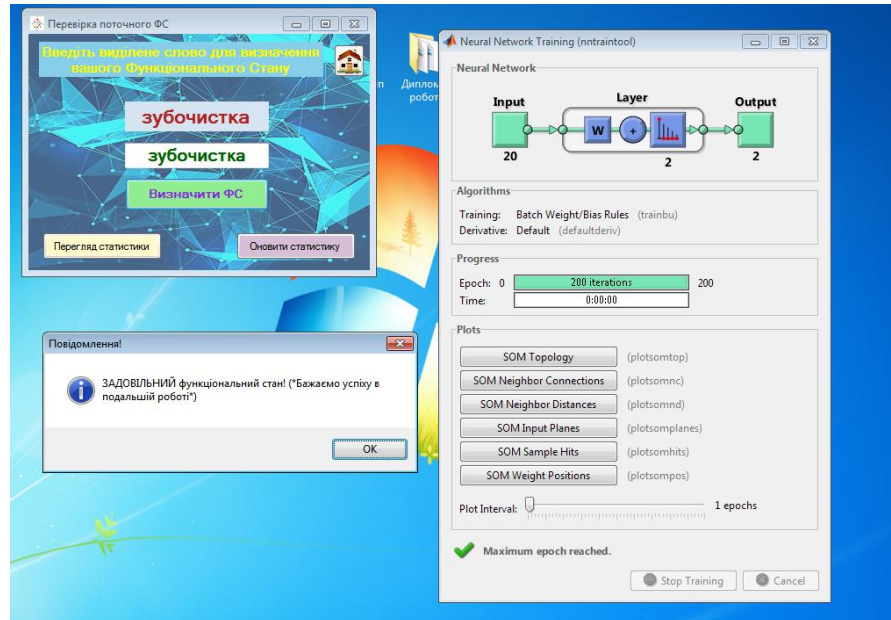


Рисунок 3.11 – Фрагмент робочого столу після визначення функціонального стану

Є можливість переглянути статистичні дані у файлі не виходячи з програми. Слід у вікні «Перевірка поточного ФС» натиснути на кнопку «Перегляд статистики». У результаті відкриється Excel-файл із результатами всіх перевірок певного користувача (рис. 3.12).

	1	2	3	4
1	05.06.2019 18:16	Задовільний функціональний стан	катастрофа	
2	05.06.2019 18:17	Незадовільний функціональний стан	дендропарк	
3	05.06.2019 18:18	Задовільний функціональний стан	зубочистка	
4				
5				
6				

Рисунок 3.12 – Перегляд файлу «Статистика»

У програмі передбачено різні виключні ситуації, всі можливі помилки при роботі з програмою. Коли користувач буде робити ту чи іншу невірну, помилкову дію або ж навпаки створюватиме, додаватиме, то отримуватиме інформаційні повідомлення різного характеру. Також передбачено спливаючі підказки, адже більшість кнопок немає написів (рис. 3.13).

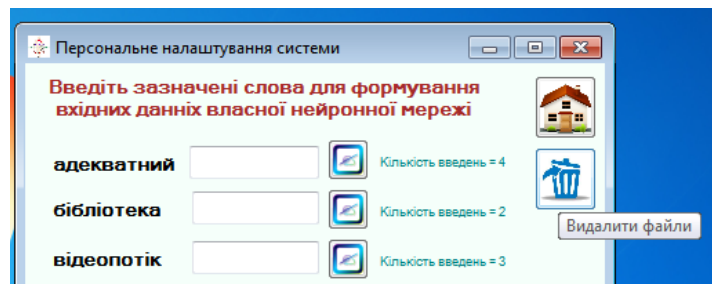


Рисунок 3.13 – Фрагмент одного з прикладів спливаючих підказок

У програмі передбачено велику кількість інформаційних повідомлень та повідомлень про помилки. Деякі з них наведені на рис. 3.14.

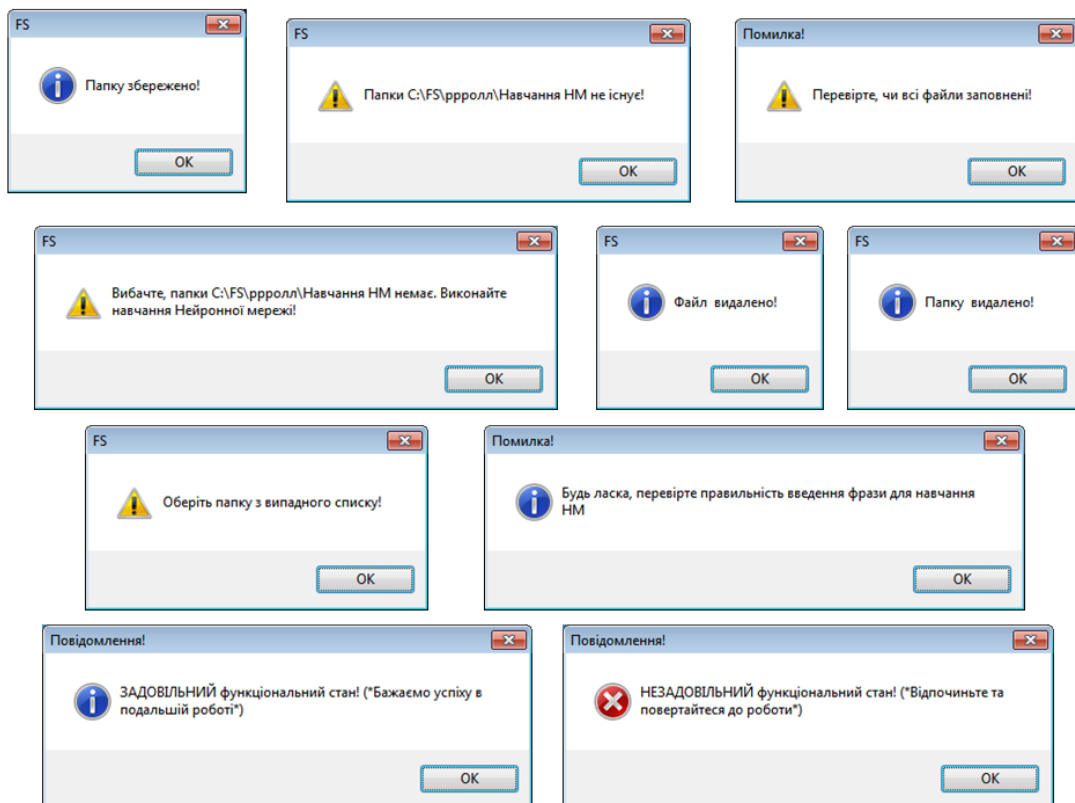


Рисунок 3.14 – Можливі повідомлення (попередження, помилки, тощо)

3.4 Висновки до розділу 3

Визначення функціонального стану оператора зручно проводити з використанням мереж Кохонена, якщо на вхід подавати імпульси, які надходять з клавіатури.

Для реалізації моделі визначення функціонального стану зручно використовувати розроблену в середовищі Visual Basic 2010, з залученням середовища MatLab R2011b, спеціальну програму.

Інформаційна система є stand-alone додатком, не потребує додаткового встановлення будь-яких девайсів.

Програма працює за принципом порівняння вектора, що надходить, з еталонним (існує спеціальна база еталонних значень) і віднесення його до одного з двох можливих кластерів.

ВИСНОВКИ

Аналіз автоматизованих виробництв виявив їх людино-машинний характер, напруженість діяльності операторів та проблему постійного контролю функціонального стану операторів.

Погане самопочуття несе ризик збоїв на підприємствах, в системах, але в першу чергу втрату або відсутність операторів на їх робочих місцях.

Біометрика дозволяє ідентифікувати та аутентифікувати людину на основі сукупності пізнаваних і перевічених даних. На відміну від паролів, значків чи документів, біометричні дані не можна забувати, обмінювати, красти або підробляти.

Для реалізації розпізнавання за клавіатурним почерком вибрано алгоритм, який перевіряє імпульси, що поступають з клавіатури, порівнюючи їх з еталонним значенням, яке створила нейронна мережа. Обрано невибагливу та максимально точну нейронну мережу Кохонена (різновид – Самоорганізована мережа Кохонена).

Для реалізації моделі визначення функціонального стану зручно використовувати розроблену в середовищі Visual Basic 2010, з залученням середовища MatLab R2011b, спеціальну програму.

Інформаційна система є stand-alone додатком, не потребує додаткового встановлення будь-яких девайсів.

Програма працює за принципом порівняння вектора, що надходить, з еталонним (існує спеціальна база еталонних значень) і віднесення його до одного з двох можливих кластерів.

Наукова новизна. На відміну від існуючих апаратних методів запропоновано невибагливий до технічного забезпечення метод, орієнтований на аналіз клавіатурного почерку з використанням нейронних мереж Кохонена.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Лавров Е.А., Скиданенко А.С. Эргономические резервы повышения эффективности АСУТП производства удобрений. // Сучасні інформаційні системи і технології: матеріали Другої міжнародної науково-практичної конференції, м.Суми, 21-24 травня 2013 р. / редкол. А.С. Довбиш, О.А. Борисенко, О.В. Бондар.-Суми: Сумський державний університет, 2013.-С.53-54.
2. Haykin S. Neural networks, a comprehensive foundation. – N.Y.: Macmillan College Publishing Company, 2000.
3. Ключарев В.А., Никишина И.С., Лысков Е.Б., Сандстрем М., Хансон Милд К. Влияние слабых электромагнитных полей на стабильность изображения компьютерного монитора: возможные последствия для операторов Текст. / Физиология человека. 2000, т.26, сч.54.
4. Иванов А. И. Нейросетевые технологии биометрической аутентификации пользователей открытых систем Текст. // Автореф. дис. . доктора техн. наук. Пенза: Изд-во Пенз. гос. ун-та, 2002.
5. Компьютерная графология Электронный ресурс. / Режим доступа . – <http://www.geocities.com/werebad/art1999/compgrafo.htm>
6. Иванов А. И. Биометрическая идентификация личности по динамике подсознательных движений [Текст]. Пенза: Изд-во Пенз. гос. ун-та, 2000.-210 с.
7. В.И.Волчихин, А.И.Иванов Естественное использование искусственных нейронных сетей в биометрии [Текст]. // "Системы безопасности" №3 2002 г. стр. 48-49.
8. Волчихин В. И. Биометрия: быстрое обучение искусственных нейронных сетей [Текст]. / В.И.Волчихин, А.И.Иванов. Пенза: Изд-во Пенз. гос. унта, 2000.- 125 с.

9. Волчихин В. И. Использование тайных биометрических образов человека [Текст]. / В.И. Волчихин, А.И. Иванов // Системы безопасности. 2002. -№2(44). -С. 40-41.
10. Галушкин А. И. Нейронные сети: история развития: Учеб. пособие для вузов [Текст]. / А. И. Галушкин, Я. З. Цыпкин // М.: ИПРЖ, 2001. 214 с.
11. Дорофеева Э.Т. О возможных критериях распознавания эмоциональных состояний [Текст]. // Проблемы моделирования психической деятельности. Вып.2 Новосибирск, 2002. с. 65-68.
12. Практикум по общей экспериментальной и прикладной психологии. Под ред. Крылова А. А, Маничева С. А.— СПб.: Питер, 2003. – 560 с.
13. Юрков Н. К. Интеллектуальные компьютерные обучающие системы: моногр./ Н.К.Юрков – Пенза:Изд-во ПГУ, 2010. – 304 с.
14. Суздальцев А.И., Лобанова В.А., Абашин В.Г. Определение психофизиологического состояния оперативного персонала по клавиатурному почерку на нефтеперерабатывающих мини-заводах//Электронный журнал «Нефтегазовое дело», 2006. /режим доступа.– http://www.ogbus.ru/authors/Suzdaltsev/Suzdaltsev_1.pdf
15. Дитхелм Герд Управление проектами. СПб, Бизнес-процесса, 2003, Том 1 "Основы", 390 с., Том 2 "Особенности", 274 с.
16. Иванов А. И. Нейросетевые технологии биометрической аутентификации пользователей открытых систем [Текст] // Автореф. дис. ... доктора техн. наук. - Пенза: Изд-во Пенз. гос. ун-та, 2002.
17. Абашин В.Г. Адаптивная математическая модель мультибиометрической подсистемы определения работоспособности человека-оператора АРМ на основе нечетких множеств. Информационные системы и технологии 2011. - №5(додаток Д).С. 90-96

ДОДАТОК А – Лістинг програмного коду

Код для головного вікна програми «Визначення ФС» (Form3)

```
Imports System
Imports System.IO
Imports System.Windows.Forms.ComboBox
Module Module1
    Public Direct As String
End Module

Public Class Form3

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        If (Not System.IO.Directory.Exists("C:\FS")) Then
            System.IO.Directory.CreateDirectory("C:\FS")
            MsgBox("Папку C:\FS створено!",
MessageBoxIcon.Information)
        End If
        For Each dir_name As String In
Directory.GetDirectories("C:\FS\")
            ComboBox1.Items.Add(dir_name)
            ComboBox2.Items.Add(dir_name)
        Next
    End Sub

    Private Sub ComboBox1_SelectedValueChanged(ByVal sender As
Object, ByVal e As System.EventArgs) Handles
ComboBox1.SelectedValueChanged
        Direct = ComboBox1.Text
        Form4.Visible = True
        Me.Visible = False
    End Sub

    Private Sub someButton_MouseHover(ByVal sender As Object,
ByVal e As System.EventArgs) _
Handles Button3.MouseHover
        Dim toolTip As New ToolTip()
        toolTip.SetToolTip(Button3, "Створити папку")
        toolTip.SetToolTip(Button4, "Видалити папку")
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button1.Click
        If TextBox1.Text = "" Then
            MsgBox("Ви не вказали назву створюваної папки!",
MessageBoxIcon.Warning)
        ElseIf (Not System.IO.Directory.Exists("C:\FS\" &
TextBox1.Text)) Then
            System.IO.Directory.CreateDirectory("C:\FS\" &
TextBox1.Text)
        End If
    End Sub
End Class
```

```

        MsgBox("Папку збережено!", MessageBoxIcon.Information)
        ComboBox1.Items.Clear()
        ComboBox2.Items.Clear()
        For Each dir_name As String In
Directory.GetDirectories("C:\FS\")
            ComboBox1.Items.Add(dir_name)
            ComboBox2.Items.Add(dir_name)
        Next
        TextBox1.Text = ""
        ElseIf MsgBox("Вибачте, але папка з такою назвою вже
існує!", MessageBoxIcon.Warning) Then
            End If

    End Sub

    Private Sub TextBox1_KeyPress(ByVal sender As Object, ByVal e
As KeyPressEventArgs) Handles TextBox1.KeyPress
        If Not Char.IsLetter(e.KeyChar) And Not
Char.IsControl(e.KeyChar) Then
            e.KeyChar = ""
        End If
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button3.Click
        Panel1.Visible = True
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button4.Click
        Panel2.Visible = True
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button2.Click
        If ComboBox2.SelectedIndex = -1 Then
            MsgBox("Оберіть папку з випадного списку!",
MessageBoxIcon.Warning)
        Else :
CreateObject("Scripting.FileSystemObject").GetFolder(ComboBox2.Text
.ToString()).Delete()
            ComboBox1.Items.Clear()
            ComboBox2.Items.Clear()
            For Each dir_name As String In
Directory.GetDirectories("C:\FS\")
                ComboBox1.Items.Add(dir_name)
                ComboBox2.Items.Add(dir_name)
            Next
            ComboBox2.Text = ""
            MsgBox("Папку " & ComboBox2.Text.ToString() & "
видалено!", MessageBoxIcon.Information)
        End If
    End Sub

```



```

End Sub

Private Sub ПроАвтораToolStripMenuItem_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
ПроАвтораToolStripMenuItem.Click
    Form6.Visible = True
    Me.Visible = False
End Sub

End Class

```

Код для проміжного вікна (Form4)

```
Imports System.IO
```

```
Public Class Form4
```

```

Private Sub Form4_Closing(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.FormClosing
    Form3.Visible = True
End Sub

```

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button1.Click
    Form1.Visible = True
    Me.Visible = False
End Sub

```

```

Private Sub Button2_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button2.Click
    If Directory.Exists(Direct) Then
        If Directory.Exists(Direct & "\Навчання НМ") Then
            If Not File.Exists(Direct & "\Навчання
НМ\адекватний_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\бібліотека_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\відеопотік_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\гермомішок_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\дендропарк_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\зубочистка_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\катастрофа_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\латифундія_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\менеджмент_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\наповнення_train.txt") Then
                MessageBox.Show("Перевірте, чи всі файли
заповнені!", "Помилка!", MessageBoxButtons.OK,
MessageBoxIcon.Warning)
            Else
                Dim m1String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\адекватний_train.txt", System.Text.Encoding.Default)
                Dim b1 = m1String.Length
                Dim m2String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\бібліотека_train.txt", System.Text.Encoding.Default)
                Dim b2 = m2String.Length

```

```

        Dim m3String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\відеопотік_train.txt", System.Text.Encoding.Default)
        Dim b3 = m3String.Length
        Dim m4String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\гермомішок_train.txt", System.Text.Encoding.Default)
        Dim b4 = m4String.Length
        Dim m5String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\дендропарк_train.txt", System.Text.Encoding.Default)
        Dim b5 = m5String.Length
        Dim m6String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\зубочистка_train.txt", System.Text.Encoding.Default)
        Dim b6 = m6String.Length
        Dim m7String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\катастрофа_train.txt", System.Text.Encoding.Default)
        Dim b7 = m7String.Length
        Dim m8String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\латифундія_train.txt", System.Text.Encoding.Default)
        Dim b8 = m8String.Length
        Dim m9String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\менеджмент_train.txt", System.Text.Encoding.Default)
        Dim b9 = m9String.Length
        Dim m10String() = IO.File.ReadAllLines(Direct
& "\Навчання НМ\наповнення_train.txt",
System.Text.Encoding.Default)
        Dim b10 = m10String.Length
        If b1 = 0 Or b2 = 0 Or b3 = 0 Or b4 = 0 Or b5
= 0 Or b6 = 0 Or b7 = 0 Or b8 = 0 Or b9 = 0 Or b10 = 0 Then
            MessageBox.Show("Перевірте, чи всі файли
заповнені!", "Помилка!", MessageBoxButtons.OK,
MessageBoxIcon.Warning)
        Else
            Form2.Visible = True
            Me.Visible = False
        End If
    End If
    ElseIf MsgBox("Вибачте, папки " & Direct & "\Навчання
НМ немає. Виконайте навчання Нейронної мережі!",
MessageBoxIcon.Warning) Then
        End If
        ElseIf MsgBox("Вибачте, папка " & Direct & " пуста.
Виконайте навчання Нейронної мережі!", MessageBoxIcon.Warning)
Then
            End If
        End Sub
    End Class

```

Код для «Персональне налаштування системи» (Form1)

```

Imports System.IO
Imports System.Runtime.InteropServices

Public Class Form1

```

```

    Dim kd1, kd2, kd3, kd4, kd5, kd6, kd7, kd8, kd9, kd10 As Date
    'событие нажатия на кнопку
    Dim ku1, ku2, ku3, ku4, ku5, ku6, ku7, ku8, ku9, ku10 As Date
    'событие отпуска кнопки
    Dim tn1, tn2, tn3, tn4, tn5, tn6, tn7, tn8, tn9, tn10 As
Single 'время между нажатиями кнопок
    Dim tu1, tu2, tu3, tu4, tu5, tu6, tu7, tu8, tu9, tu10 As
Single 'время удержания кнопок
    Dim reader As StreamReader

```

```

    Private Sub TextBox1_KeyDown(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TextBox1.KeyDown

```

```

    If e.KeyCode = 70 Then
        kd1 = Now()
    End If
    If e.KeyCode = 76 Then
        kd2 = Now()
    End If
    If e.KeyCode = 84 Then
        kd3 = Now()
    End If
    If e.KeyCode = 82 Then
        kd4 = Now()
    End If
    If e.KeyCode = 68 Then
        kd5 = Now()
    End If
    If e.KeyCode = 70 Then
        kd6 = Now()
    End If
    If e.KeyCode = 78 Then
        kd7 = Now()
    End If
    If e.KeyCode = 89 Then
        kd8 = Now()
    End If
    If e.KeyCode = 66 Then
        kd9 = Now()
    End If
    If e.KeyCode = 81 Then
        kd10 = Now()
    End If

```

```
End Sub
```

```
\.....
```

```

    Private Sub TextBox10_KeyUp(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TextBox10.KeyUp

```

```

    If e.KeyCode = 89 Then
        ku1 = Now()
    End If
    If e.KeyCode = 70 Then

```

```

        ku2 = Now()
    End If
    If e.KeyCode = 71 Then
        ku3 = Now()
    End If
    If e.KeyCode = 74 Then
        ku4 = Now()
    End If
    If e.KeyCode = 68 Then
        ku5 = Now()
    End If
    If e.KeyCode = 89 Then
        ku6 = Now()
    End If
    If e.KeyCode = 84 Then
        ku7 = Now()
    End If
    If e.KeyCode = 89 Then
        ku8 = Now()
    End If
    If e.KeyCode = 89 Then
        ku9 = Now()
    End If
    If e.KeyCode = 90 Then
        ku10 = Now()
    End If
End Sub

```

```

Private Sub Form1_Closed(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.FormClosed
    Form4.Visible = True
End Sub

```

```

Private Sub someButton_MouseHover(ByVal sender As Object,
ByVal e As System.EventArgs) _
Handles Button1.MouseHover
    Dim toolTip As New ToolTip()
    toolTip.SetToolTip(Button1, "Додати в файл")
    toolTip.SetToolTip(Button2, "Визначити поточний
функціональний стан")
    toolTip.SetToolTip(Button3, "На головну")
    toolTip.SetToolTip(Button4, "Додати в файл")
    toolTip.SetToolTip(Button5, "Додати в файл")
    toolTip.SetToolTip(Button6, "Додати в файл")
    toolTip.SetToolTip(Button7, "Додати в файл")
    toolTip.SetToolTip(Button8, "Додати в файл")
    toolTip.SetToolTip(Button9, "Додати в файл")
    toolTip.SetToolTip(Button10, "Додати в файл")
    toolTip.SetToolTip(Button11, "Додати в файл")
    toolTip.SetToolTip(Button12, "Додати в файл")
    toolTip.SetToolTip(Button13, "Видалити файли")
End Sub

```

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button1.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання HM")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання HM")
        ElseIf (Not File.Exists(Direct & "\Навчання HM\" &
Label1.Text & "_train.txt")) Then
            Dim g
            g = File.Create(Direct & "\Навчання HM\" & Label1.Text
& "_train.txt")
            g.Close()
        End If
        Try
            Dim mlString() = IO.File.ReadAllLines(Direct &
"\Навчання HM\" & Label1.Text & "_train.txt",
System.Text.Encoding.Default)
            Dim bl = mlString.Length
            If TextBox1.Text = Label1.Text Then
                tn1 = 0
                tu1 = (ku1 - kd1).Milliseconds / 1000
                tn2 = (kd2 - ku1).Milliseconds / 1000
                tu2 = (ku2 - kd2).Milliseconds / 1000
                tn3 = (kd3 - ku2).Milliseconds / 1000
                tu3 = (ku3 - kd3).Milliseconds / 1000
                tn4 = (kd4 - ku3).Milliseconds / 1000
                tu4 = (ku4 - kd4).Milliseconds / 1000
                tn5 = (kd5 - ku4).Milliseconds / 1000
                tu5 = (ku5 - kd5).Milliseconds / 1000
                tn6 = (kd6 - ku5).Milliseconds / 1000
                tu6 = (ku6 - kd6).Milliseconds / 1000
                tn7 = (kd7 - ku6).Milliseconds / 1000
                tu7 = (ku7 - kd7).Milliseconds / 1000
                tn8 = (kd8 - ku7).Milliseconds / 1000
                tu8 = (ku8 - kd8).Milliseconds / 1000
                tn9 = (kd9 - ku8).Milliseconds / 1000
                tu9 = (ku9 - kd9).Milliseconds / 1000
                tn10 = (kd10 - ku9).Milliseconds / 1000
                tu10 = (ku10 - kd10).Milliseconds / 1000
                Dim writer As New StreamWriter(Direct & "\Навчання
HM\" & Label1.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
                writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст
                writer.Close() ' Закриваємо файл
                Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання HM\" & Label1.Text & "_train.txt",
System.Text.Encoding.Default)
                Txt = Txt.Replace(", ", ".")
            End If
        Catch ex As Exception
            MsgBox(ex.Message)
        End Try
    End If
End Sub

```

```

        IO.File.WriteAllText(Direct & "\Навчання НМ\" &
Label11.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label12.Text = "Кількість введень = " & b1 + 1
        TextBox1.Text = ""
        TextBox1.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання НМ", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
    If Not File.Exists(Direct & "\Навчання НМ\" & Label1.Text
& "_train.txt") Then
        Label12.Text = "Кількість введень = 0"
    Else : Dim m1String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label1.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b1 = m1String.Length
        Label12.Text = "Кількість введень = " & b1
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label2.Text
& "_train.txt") Then
        Label13.Text = "Кількість введень = 0"
    Else : Dim m2String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label2.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b2 = m2String.Length
        Label13.Text = "Кількість введень = " & b2
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label3.Text
& "_train.txt") Then
        Label14.Text = "Кількість введень = 0"
    Else : Dim m3String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label3.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b3 = m3String.Length
        Label14.Text = "Кількість введень = " & b3
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label4.Text
& "_train.txt") Then
        Label15.Text = "Кількість введень = 0"
    Else : Dim m4String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label4.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b4 = m4String.Length
        Label15.Text = "Кількість введень = " & b4
    End If
End Sub

```

```

    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label5.Text
& "_train.txt") Then
        Label16.Text = "Кількість введень = 0"
    Else : Dim m5String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label5.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b5 = m5String.Length
        Label16.Text = "Кількість введень = " & b5
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label6.Text
& "_train.txt") Then
        Label17.Text = "Кількість введень = 0"
    Else : Dim m6String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label6.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b6 = m6String.Length
        Label17.Text = "Кількість введень = " & b6
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label7.Text
& "_train.txt") Then
        Label18.Text = "Кількість введень = 0"
    Else : Dim m7String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label7.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b7 = m7String.Length
        Label18.Text = "Кількість введень = " & b7
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label8.Text
& "_train.txt") Then
        Label19.Text = "Кількість введень = 0"
    Else : Dim m8String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label8.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b8 = m8String.Length
        Label19.Text = "Кількість введень = " & b8
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label9.Text
& "_train.txt") Then
        Label20.Text = "Кількість введень = 0"
    Else : Dim m9String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label9.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b9 = m9String.Length
        Label20.Text = "Кількість введень = " & b9
    End If
    If Not File.Exists(Direct & "\Навчання НМ\" & Label10.Text
& "_train.txt") Then
        Label21.Text = "Кількість введень = 0"
    Else : Dim m10String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label10.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b10 = m10String.Length

```

```

        Label21.Text = "Кількість введень = " & b10
    End If
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button2.Click ' FS
    If Directory.Exists(Direct) Then
        If Directory.Exists(Direct & "\Навчання НМ") Then
            If Not File.Exists(Direct & "\Навчання
НМ\адекватний_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\бібліотека_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\відеопотік_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\гермомішок_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\дендропарк_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\зубочистка_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\катастрофа_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\латифундія_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\менеджмент_train.txt") Or Not File.Exists(Direct & "\Навчання
НМ\наповнення_train.txt") Then
                MessageBox.Show("Перевірте, чи всі файли
заповнені!", "Помилка!", MessageBoxButtons.OK,
MessageBoxIcon.Warning)
            Else
                Dim m1String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\адекватний_train.txt", System.Text.Encoding.Default)
                Dim b1 = m1String.Length
                Dim m2String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\бібліотека_train.txt", System.Text.Encoding.Default)
                Dim b2 = m2String.Length
                Dim m3String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\відеопотік_train.txt", System.Text.Encoding.Default)
                Dim b3 = m3String.Length
                Dim m4String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\гермомішок_train.txt", System.Text.Encoding.Default)
                Dim b4 = m4String.Length
                Dim m5String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\дендропарк_train.txt", System.Text.Encoding.Default)
                Dim b5 = m5String.Length
                Dim m6String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\зубочистка_train.txt", System.Text.Encoding.Default)
                Dim b6 = m6String.Length
                Dim m7String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\катастрофа_train.txt", System.Text.Encoding.Default)
                Dim b7 = m7String.Length
                Dim m8String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\латифундія_train.txt", System.Text.Encoding.Default)
                Dim b8 = m8String.Length
                Dim m9String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\менеджмент_train.txt", System.Text.Encoding.Default)
                Dim b9 = m9String.Length
                Dim m10String() = IO.File.ReadAllLines(Direct
& "\Навчання НМ\наповнення_train.txt",
System.Text.Encoding.Default)

```



```

        Dim b10 = m10String.Length
        If b1 = 0 Or b2 = 0 Or b3 = 0 Or b4 = 0 Or b5
= 0 Or b6 = 0 Or b7 = 0 Or b8 = 0 Or b9 = 0 Or b10 = 0 Then
            MessageBox.Show("Перевірте, чи всі файли
заповнені!", "Помилка!", MessageBoxButtons.OK,
MessageBoxIcon.Warning)
        Else
            Form2.Visible = True
            Me.Visible = False
        End If
    End If
    ElseIf MessageBox.Show("Вибачте, папки " & Direct &
"\Навчання НМ немає. Виконайте навчання Нейронної мережі!",
"Помилка!", MessageBoxButtons.OK, MessageBoxIcon.Warning) Then
        End If
    ElseIf MessageBox.Show("Вибачте, папка " & Direct & "
пуста. Виконайте навчання Нейронної мережі!", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Warning) Then
        End If
    End Sub

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button3.Click 'main
        Form3.Visible = True
        Me.Visible = False
    End Sub

    Private Sub Button4_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button4.Click
        If Not System.IO.Directory.Exists(Direct & "\Навчання НМ")
Then
            System.IO.Directory.CreateDirectory(Direct &
"\Навчання НМ")
        ElseIf (Not File.Exists(Direct & "\Навчання НМ\" &
Label2.Text & "_train.txt")) Then
            Dim g
            g = File.Create(Direct & "\Навчання НМ\" & Label2.Text
& "_train.txt")
            g.Close()
        End If
        Try
            Dim m2String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label2.Text & "_train.txt",
System.Text.Encoding.Default)
            Dim b2 = m2String.Length
            If TextBox2.Text = Label2.Text Then
                tn1 = 0
                tu1 = (ku1 - kd1).Milliseconds / 1000
                tn2 = (kd2 - ku1).Milliseconds / 1000
                tu2 = (ku2 - kd2).Milliseconds / 1000
                tn3 = (kd3 - ku2).Milliseconds / 1000
                tu3 = (ku3 - kd3).Milliseconds / 1000
                tn4 = (kd4 - ku3).Milliseconds / 1000
            End If
        End Try
    End Sub

```

```

        tu4 = (ku4 - kd4).Milliseconds / 1000
        tn5 = (kd5 - ku4).Milliseconds / 1000
        tu5 = (ku5 - kd5).Milliseconds / 1000
        tn6 = (kd6 - ku5).Milliseconds / 1000
        tu6 = (ku6 - kd6).Milliseconds / 1000
        tn7 = (kd7 - ku6).Milliseconds / 1000
        tu7 = (ku7 - kd7).Milliseconds / 1000
        tn8 = (kd8 - ku7).Milliseconds / 1000
        tu8 = (ku8 - kd8).Milliseconds / 1000
        tn9 = (kd9 - ku8).Milliseconds / 1000
        tu9 = (ku9 - kd9).Milliseconds / 1000
        tn10 = (kd10 - ku9).Milliseconds / 1000
        tu10 = (ku10 - kd10).Milliseconds / 1000
        Dim writer As New StreamWriter(Direct & "\Навчання
HM\" & Label2.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
        writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст
        writer.Close() ' Закриваємо файл
        Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання HM\" & Label2.Text & "_train.txt",
System.Text.Encoding.Default)
        Txt = Txt.Replace(",", ".")
        IO.File.WriteAllText(Direct & "\Навчання HM\" &
Label2.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label13.Text = "Кількість введень = " & b2 + 1
        TextBox2.Text = ""
        TextBox2.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання HM", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button5_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button5.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання HM")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання HM")
    ElseIf (Not File.Exists(Direct & "\Навчання HM\" &
Label3.Text & "_train.txt")) Then
        Dim g

```

```

        g = File.Create(Direct & "\Навчання НМ\" & Label3.Text
& "_train.txt")
        g.Close()
    End If
    Try
        Dim m3String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label3.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b3 = m3String.Length
        If TextBox3.Text = Label3.Text Then
            tn1 = 0
            tu1 = (ku1 - kd1).Milliseconds / 1000
            tn2 = (kd2 - ku1).Milliseconds / 1000
            tu2 = (ku2 - kd2).Milliseconds / 1000
            tn3 = (kd3 - ku2).Milliseconds / 1000
            tu3 = (ku3 - kd3).Milliseconds / 1000
            tn4 = (kd4 - ku3).Milliseconds / 1000
            tu4 = (ku4 - kd4).Milliseconds / 1000
            tn5 = (kd5 - ku4).Milliseconds / 1000
            tu5 = (ku5 - kd5).Milliseconds / 1000
            tn6 = (kd6 - ku5).Milliseconds / 1000
            tu6 = (ku6 - kd6).Milliseconds / 1000
            tn7 = (kd7 - ku6).Milliseconds / 1000
            tu7 = (ku7 - kd7).Milliseconds / 1000
            tn8 = (kd8 - ku7).Milliseconds / 1000
            tu8 = (ku8 - kd8).Milliseconds / 1000
            tn9 = (kd9 - ku8).Milliseconds / 1000
            tu9 = (ku9 - kd9).Milliseconds / 1000
            tn10 = (kd10 - ku9).Milliseconds / 1000
            tu10 = (ku10 - kd10).Milliseconds / 1000
            Dim writer As New StreamWriter(Direct & "\Навчання
НМ\" & Label3.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
            writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст

            writer.Close() ' Закриваємо файл
            Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання НМ\" & Label3.Text & "_train.txt",
System.Text.Encoding.Default)
            Txt = Txt.Replace(", ", ".")
            IO.File.WriteAllText(Direct & "\Навчання НМ\" &
Label3.Text & "_train.txt", Txt, System.Text.Encoding.Default)

            Label14.Text = "Кількість введень = " & b3 + 1
            TextBox3.Text = ""
            TextBox3.Select()
        Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання НМ", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)

```

```

        End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button6_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button6.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання HM")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання HM")
    ElseIf (Not File.Exists(Direct & "\Навчання HM\" &
Label4.Text & "_train.txt")) Then
        Dim g
        g = File.Create(Direct & "\Навчання HM\" & Label4.Text
& "_train.txt")
        g.Close()
    End If
    Try
        Dim m4String() = IO.File.ReadAllLines(Direct &
"\Навчання HM\" & Label4.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b4 = m4String.Length
        If TextBox4.Text = Label4.Text Then
            tn1 = 0
            tu1 = (ku1 - kd1).Milliseconds / 1000
            tn2 = (kd2 - ku1).Milliseconds / 1000
            tu2 = (ku2 - kd2).Milliseconds / 1000
            tn3 = (kd3 - ku2).Milliseconds / 1000
            tu3 = (ku3 - kd3).Milliseconds / 1000
            tn4 = (kd4 - ku3).Milliseconds / 1000
            tu4 = (ku4 - kd4).Milliseconds / 1000
            tn5 = (kd5 - ku4).Milliseconds / 1000
            tu5 = (ku5 - kd5).Milliseconds / 1000
            tn6 = (kd6 - ku5).Milliseconds / 1000
            tu6 = (ku6 - kd6).Milliseconds / 1000
            tn7 = (kd7 - ku6).Milliseconds / 1000
            tu7 = (ku7 - kd7).Milliseconds / 1000
            tn8 = (kd8 - ku7).Milliseconds / 1000
            tu8 = (ku8 - kd8).Milliseconds / 1000
            tn9 = (kd9 - ku8).Milliseconds / 1000
            tu9 = (ku9 - kd9).Milliseconds / 1000
            tn10 = (kd10 - ku9).Milliseconds / 1000
            tu10 = (ku10 - kd10).Milliseconds / 1000
            Dim writer As New StreamWriter(Direct & "\Навчання
HM\" & Label4.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
            writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &

```

```

tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст
        writer.Close() ' Закриваємо файл
        Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання НМ\" & Label4.Text & "_train.txt",
System.Text.Encoding.Default)
        Txt = Txt.Replace(",", ".")
        IO.File.WriteAllText(Direct & "\Навчання НМ\" &
Label4.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label15.Text = "Кількість введень = " & b4 + 1
        TextBox4.Text = ""
        TextBox4.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання НМ", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button7_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button7.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання НМ")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання НМ")
    ElseIf (Not File.Exists(Direct & "\Навчання НМ\" &
Label5.Text & "_train.txt")) Then
        Dim g
        g = File.Create(Direct & "\Навчання НМ\" & Label5.Text
& "_train.txt")
        g.Close()
    End If
    Try
        Dim m5String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label5.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b5 = m5String.Length
        If TextBox5.Text = Label5.Text Then
            tn1 = 0
            tu1 = (ku1 - kd1).Milliseconds / 1000
            tn2 = (kd2 - ku1).Milliseconds / 1000
            tu2 = (ku2 - kd2).Milliseconds / 1000
            tn3 = (kd3 - ku2).Milliseconds / 1000
            tu3 = (ku3 - kd3).Milliseconds / 1000
            tn4 = (kd4 - ku3).Milliseconds / 1000
            tu4 = (ku4 - kd4).Milliseconds / 1000
            tn5 = (kd5 - ku4).Milliseconds / 1000
            tu5 = (ku5 - kd5).Milliseconds / 1000
            tn6 = (kd6 - ku5).Milliseconds / 1000

```

```

        tu6 = (ku6 - kd6).Milliseconds / 1000
        tn7 = (kd7 - ku6).Milliseconds / 1000
        tu7 = (ku7 - kd7).Milliseconds / 1000
        tn8 = (kd8 - ku7).Milliseconds / 1000
        tu8 = (ku8 - kd8).Milliseconds / 1000
        tn9 = (kd9 - ku8).Milliseconds / 1000
        tu9 = (ku9 - kd9).Milliseconds / 1000
        tn10 = (kd10 - ku9).Milliseconds / 1000
        tu10 = (ku10 - kd10).Milliseconds / 1000
        Dim writer As New StreamWriter(Direct & "\Навчання
HM\" & Label5.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
        writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст

        writer.Close() ' Закриваємо файл
        Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання HM\" & Label5.Text & "_train.txt",
System.Text.Encoding.Default)
        Txt = Txt.Replace(",", ".")
        IO.File.WriteAllText(Direct & "\Навчання HM\" &
Label5.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label16.Text = "Кількість введень = " & b5 + 1
        TextBox5.Text = ""
        TextBox5.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання HM", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button8_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button8.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання HM")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання HM")
    ElseIf (Not File.Exists(Direct & "\Навчання HM\" &
Label6.Text & "_train.txt")) Then
        Dim g
        g = File.Create(Direct & "\Навчання HM\" & Label6.Text
& "_train.txt")
        g.Close()
    End If
    Try

```

```

        Dim m6String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label6.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b6 = m6String.Length
        If TextBox6.Text = Label6.Text Then
            tn1 = 0
            tu1 = (ku1 - kd1).Milliseconds / 1000
            tn2 = (kd2 - ku1).Milliseconds / 1000
            tu2 = (ku2 - kd2).Milliseconds / 1000
            tn3 = (kd3 - ku2).Milliseconds / 1000
            tu3 = (ku3 - kd3).Milliseconds / 1000
            tn4 = (kd4 - ku3).Milliseconds / 1000
            tu4 = (ku4 - kd4).Milliseconds / 1000
            tn5 = (kd5 - ku4).Milliseconds / 1000
            tu5 = (ku5 - kd5).Milliseconds / 1000
            tn6 = (kd6 - ku5).Milliseconds / 1000
            tu6 = (ku6 - kd6).Milliseconds / 1000
            tn7 = (kd7 - ku6).Milliseconds / 1000
            tu7 = (ku7 - kd7).Milliseconds / 1000
            tn8 = (kd8 - ku7).Milliseconds / 1000
            tu8 = (ku8 - kd8).Milliseconds / 1000
            tn9 = (kd9 - ku8).Milliseconds / 1000
            tu9 = (ku9 - kd9).Milliseconds / 1000
            tn10 = (kd10 - ku9).Milliseconds / 1000
            tu10 = (ku10 - kd10).Milliseconds / 1000
            Dim writer As New StreamWriter(Direct & "\Навчання
НМ\" & Label6.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
                writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст
                writer.Close() ' Закриваємо файл
            Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання НМ\" & Label6.Text & "_train.txt",
System.Text.Encoding.Default)
                Txt = Txt.Replace(",", ".")
                IO.File.WriteAllText(Direct & "\Навчання НМ\" &
Label6.Text & "_train.txt", Txt, System.Text.Encoding.Default)

                Label17.Text = "Кількість введень = " & b6 + 1
                TextBox6.Text = ""
                TextBox6.Select()
        Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання НМ", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
        End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try

```

```

End Sub

Private Sub Button9_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button9.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання HM")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання HM")
        ElseIf (Not File.Exists(Direct & "\Навчання HM\" &
Label7.Text & "_train.txt")) Then
            Dim g
            g = File.Create(Direct & "\Навчання HM\" & Label7.Text
& "_train.txt")
            g.Close()
        End If
        Try
            Dim m7String() = IO.File.ReadAllLines(Direct &
"\Навчання HM\" & Label7.Text & "_train.txt",
System.Text.Encoding.Default)
            Dim b7 = m7String.Length
            If TextBox7.Text = Label7.Text Then
                tn1 = 0
                tu1 = (ku1 - kd1).Milliseconds / 1000
                tn2 = (kd2 - ku1).Milliseconds / 1000
                tu2 = (ku2 - kd2).Milliseconds / 1000
                tn3 = (kd3 - ku2).Milliseconds / 1000
                tu3 = (ku3 - kd3).Milliseconds / 1000
                tn4 = (kd4 - ku3).Milliseconds / 1000
                tu4 = (ku4 - kd4).Milliseconds / 1000
                tn5 = (kd5 - ku4).Milliseconds / 1000
                tu5 = (ku5 - kd5).Milliseconds / 1000
                tn6 = (kd6 - ku5).Milliseconds / 1000
                tu6 = (ku6 - kd6).Milliseconds / 1000
                tn7 = (kd7 - ku6).Milliseconds / 1000
                tu7 = (ku7 - kd7).Milliseconds / 1000
                tn8 = (kd8 - ku7).Milliseconds / 1000
                tu8 = (ku8 - kd8).Milliseconds / 1000
                tn9 = (kd9 - ku8).Milliseconds / 1000
                tu9 = (ku9 - kd9).Milliseconds / 1000
                tn10 = (kd10 - ku9).Milliseconds / 1000
                tu10 = (ku10 - kd10).Milliseconds / 1000
                Dim writer As New StreamWriter(Direct & "\Навчання
HM\" & Label7.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
                writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст
                writer.Close() ' Закриваємо файл
            End If
        End Try
    End If
End Sub

```



```

        Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання НМ\" & Label7.Text & "_train.txt",
System.Text.Encoding.Default)
        Txt = Txt.Replace(",", ".")
        IO.File.WriteAllText(Direct & "\Навчання НМ\" &
Label7.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label18.Text = "Кількість введень = " & b7 + 1
        TextBox7.Text = ""
        TextBox7.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання НМ", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
Catch ex As Exception
    If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button10_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles Button10.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання НМ")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання НМ")
    ElseIf (Not File.Exists(Direct & "\Навчання НМ\" &
Label8.Text & "_train.txt")) Then
        Dim g
        g = File.Create(Direct & "\Навчання НМ\" & Label8.Text
& "_train.txt")
        g.Close()
    End If
    Try
        Dim m8String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label8.Text & "_train.txt",
System.Text.Encoding.Default)
        Dim b8 = m8String.Length
        If TextBox8.Text = Label8.Text Then
            tn1 = 0
            tu1 = (ku1 - kd1).Milliseconds / 1000
            tn2 = (kd2 - ku1).Milliseconds / 1000
            tu2 = (ku2 - kd2).Milliseconds / 1000
            tn3 = (kd3 - ku2).Milliseconds / 1000
            tu3 = (ku3 - kd3).Milliseconds / 1000
            tn4 = (kd4 - ku3).Milliseconds / 1000
            tu4 = (ku4 - kd4).Milliseconds / 1000
            tn5 = (kd5 - ku4).Milliseconds / 1000
            tu5 = (ku5 - kd5).Milliseconds / 1000
            tn6 = (kd6 - ku5).Milliseconds / 1000
            tu6 = (ku6 - kd6).Milliseconds / 1000
            tn7 = (kd7 - ku6).Milliseconds / 1000
            tu7 = (ku7 - kd7).Milliseconds / 1000

```

```

        tn8 = (kd8 - ku7).Milliseconds / 1000
        tu8 = (ku8 - kd8).Milliseconds / 1000
        tn9 = (kd9 - ku8).Milliseconds / 1000
        tu9 = (ku9 - kd9).Milliseconds / 1000
        tn10 = (kd10 - ku9).Milliseconds / 1000
        tu10 = (ku10 - kd10).Milliseconds / 1000
        Dim writer As New StreamWriter(Direct & "\Навчання
HM\" & Label8.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
        writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст
        writer.Close() ' Закриваємо файл
        Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання HM\" & Label8.Text & "_train.txt",
System.Text.Encoding.Default)
        Txt = Txt.Replace(", ", ".")
        IO.File.WriteAllText(Direct & "\Навчання HM\" &
Label8.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label19.Text = "Кількість введень = " & b8 + 1
        TextBox8.Text = ""
        TextBox8.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання HM", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button11_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles Button11.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання HM")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання HM")
        ElseIf (Not File.Exists(Direct & "\Навчання HM\" &
Label9.Text & "_train.txt")) Then
            Dim g
            g = File.Create(Direct & "\Навчання HM\" & Label9.Text
& "_train.txt")
            g.Close()
        End If
        Try
            Dim m9String() = IO.File.ReadAllLines(Direct &
"\Навчання HM\" & Label9.Text & "_train.txt",
System.Text.Encoding.Default)

```

```

Dim b9 = m9String.Length
If TextBox9.Text = Label9.Text Then
    tn1 = 0
    tu1 = (ku1 - kd1).Milliseconds / 1000
    tn2 = (kd2 - ku1).Milliseconds / 1000
    tu2 = (ku2 - kd2).Milliseconds / 1000
    tn3 = (kd3 - ku2).Milliseconds / 1000
    tu3 = (ku3 - kd3).Milliseconds / 1000
    tn4 = (kd4 - ku3).Milliseconds / 1000
    tu4 = (ku4 - kd4).Milliseconds / 1000
    tn5 = (kd5 - ku4).Milliseconds / 1000
    tu5 = (ku5 - kd5).Milliseconds / 1000
    tn6 = (kd6 - ku5).Milliseconds / 1000
    tu6 = (ku6 - kd6).Milliseconds / 1000
    tn7 = (kd7 - ku6).Milliseconds / 1000
    tu7 = (ku7 - kd7).Milliseconds / 1000
    tn8 = (kd8 - ku7).Milliseconds / 1000
    tu8 = (ku8 - kd8).Milliseconds / 1000
    tn9 = (kd9 - ku8).Milliseconds / 1000
    tu9 = (ku9 - kd9).Milliseconds / 1000
    tn10 = (kd10 - ku9).Milliseconds / 1000
    tu10 = (ku10 - kd10).Milliseconds / 1000
    Dim writer As New StreamWriter(Direct & "\Навчання
HM\" & Label9.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
        writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст

        writer.Close() ' Закриваємо файл
        Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання HM\" & Label9.Text & "_train.txt",
System.Text.Encoding.Default)
        Txt = Txt.Replace(", ", ".")
        IO.File.WriteAllText(Direct & "\Навчання HM\" &
Label9.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label20.Text = "Кількість введень = " & b9 + 1
        TextBox9.Text = ""
        TextBox9.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання HM", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

```

```

Private Sub Button12_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles Button12.Click
    If Not System.IO.Directory.Exists(Direct & "\Навчання НМ")
Then
        System.IO.Directory.CreateDirectory(Direct &
"\Навчання НМ")
        ElseIf (Not File.Exists(Direct & "\Навчання НМ\" &
Label10.Text & "_train.txt")) Then
            Dim g
            g = File.Create(Direct & "\Навчання НМ\" &
Label10.Text & "_train.txt")
            g.Close()
        End If
        Try
            Dim m10String() = IO.File.ReadAllLines(Direct &
"\Навчання НМ\" & Label10.Text & "_train.txt",
System.Text.Encoding.Default)
            Dim b10 = m10String.Length
            If TextBox10.Text = Label10.Text Then
                tn1 = 0
                tu1 = (ku1 - kd1).Milliseconds / 1000
                tn2 = (kd2 - ku1).Milliseconds / 1000
                tu2 = (ku2 - kd2).Milliseconds / 1000
                tn3 = (kd3 - ku2).Milliseconds / 1000
                tu3 = (ku3 - kd3).Milliseconds / 1000
                tn4 = (kd4 - ku3).Milliseconds / 1000
                tu4 = (ku4 - kd4).Milliseconds / 1000
                tn5 = (kd5 - ku4).Milliseconds / 1000
                tu5 = (ku5 - kd5).Milliseconds / 1000
                tn6 = (kd6 - ku5).Milliseconds / 1000
                tu6 = (ku6 - kd6).Milliseconds / 1000
                tn7 = (kd7 - ku6).Milliseconds / 1000
                tu7 = (ku7 - kd7).Milliseconds / 1000
                tn8 = (kd8 - ku7).Milliseconds / 1000
                tu8 = (ku8 - kd8).Milliseconds / 1000
                tn9 = (kd9 - ku8).Milliseconds / 1000
                tu9 = (ku9 - kd9).Milliseconds / 1000
                tn10 = (kd10 - ku9).Milliseconds / 1000
                tu10 = (ku10 - kd10).Milliseconds / 1000
                Dim writer As New StreamWriter(Direct & "\Навчання
НМ\" & Label10.Text & "_train.txt", True,
System.Text.Encoding.GetEncoding(1251))
                writer.WriteLine(tn1 & vbTab & tu1 & vbTab & tn2 &
vbTab & tu2 & vbTab & tn3 & vbTab & tu3 & vbTab & tn4 & vbTab &
tu4 & vbTab & tn5 & vbTab & tu5 & vbTab & tn6 & vbTab & tu6 &
vbTab & tn7 & vbTab & tu7 & vbTab & tn8 & vbTab & tu8 & vbTab &
tn9 & vbTab & tu9 & vbTab & tn10 & vbTab & tu10) ' Записуємо в
файл текст
                writer.Close() ' Закриваємо файл
                Dim Txt As String = IO.File.ReadAllText(Direct &
"\Навчання НМ\" & Label10.Text & "_train.txt",
System.Text.Encoding.Default)
                Txt = Txt.Replace(", ", ".")
            End Try
        End If
    End If
End Sub

```

```

        IO.File.WriteAllText(Direct & "\Навчання НМ\" &
Label10.Text & "_train.txt", Txt, System.Text.Encoding.Default)

        Label21.Text = "Кількість введень = " & b10 + 1
        TextBox10.Text = ""
        TextBox10.Select()
    Else : MessageBox.Show("Будь ласка, перевірте
правильність введення фрази для навчання НМ", "Помилка!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
    End If
    Catch ex As Exception
        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button13_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles Button13.Click
    Form5.Visible = True
    Me.Visible = False
End Sub
End Class

```

Код для «Перевірка поточного фС» (Form2)

```

Imports System.IO
Imports System.Runtime.InteropServices
Imports Microsoft.Office.Interop.Excel

Public Class Form2
    Dim kd1, kd2, kd3, kd4, kd5, kd6, kd7, kd8, kd9, kd10 As Date
    'событие нажатия на кнопку
    Dim ku1, ku2, ku3, ku4, ku5, ku6, ku7, ku8, ku9, ku10 As Date
    'событие отпуска кнопки
    Dim tn1, tn2, tn3, tn4, tn5, tn6, tn7, tn8, tn9, tn10 As
Single 'время между нажатиями кнопок
    Dim tu1, tu2, tu3, tu4, tu5, tu6, tu7, tu8, tu9, tu10 As
Single 'время удержания кнопок
    Dim reader As StreamReader
    Dim Words() As String = {"адекватний", "бібліотека",
"відеопотік", "гермомішок", "дендропарк", "зубочистка",
"катастрофа", "латифундія", "менеджмент", "наповнення"}
    Dim rnd As Random = New Random

    Private Function GetNext(ByVal S() As String) As String
        Return S(rnd.Next(S.Count))
    End Function

    Private Sub TextBox1_KeyDown(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TextBox1.KeyDown
        If Label1.Text = "адекватний" Then

```

```

    If e.KeyCode = 70 Then
        kd1 = Now()
    End If
    If e.KeyCode = 76 Then
        kd2 = Now()
    End If
    If e.KeyCode = 84 Then
        kd3 = Now()
    End If
    If e.KeyCode = 82 Then
        kd4 = Now()
    End If
    If e.KeyCode = 68 Then
        kd5 = Now()
    End If
    If e.KeyCode = 70 Then
        kd6 = Now()
    End If
    If e.KeyCode = 78 Then
        kd7 = Now()
    End If
    If e.KeyCode = 89 Then
        kd8 = Now()
    End If
    If e.KeyCode = 66 Then
        kd9 = Now()
    End If
    If e.KeyCode = 81 Then
        kd10 = Now()
    End If
End If
\.....
End Sub

Private Sub TextBox1_KeyUp(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.KeyEventArgs) Handles
TextBox1.KeyUp
    If Label1.Text = "адекватный" Then
        If e.KeyCode = 70 Then
            ku1 = Now()
        End If
        If e.KeyCode = 76 Then
            ku2 = Now()
        End If
        If e.KeyCode = 84 Then
            ku3 = Now()
        End If
        If e.KeyCode = 82 Then
            ku4 = Now()
        End If
        If e.KeyCode = 68 Then
            ku5 = Now()
        End If
    End If

```

```

        If e.KeyCode = 70 Then
            ku6 = Now()
        End If
        If e.KeyCode = 78 Then
            ku7 = Now()
        End If
        If e.KeyCode = 89 Then
            ku8 = Now()
        End If
        If e.KeyCode = 66 Then
            ku9 = Now()
        End If
        If e.KeyCode = 81 Then
            ku10 = Now()
        End If
    End If
\.....
End Sub

Private Sub someButton_MouseHover(ByVal sender As Object,
ByVal e As System.EventArgs) _
    Handles Button4.MouseHover
    Dim toolTip As New ToolTip()
    toolTip.SetToolTip(Button3, "На ГОЛОВНУ")
End Sub

Private Sub Form2_FormClosed(ByVal sender As System.Object,
ByVal e As System.Windows.Forms.FormClosedEventArgs) Handles
MyBase.FormClosed
    Me.Visible = False
    Form4.Visible = True
End Sub

Private Sub Form1_Shown(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Shown
    Label1.Text = GetNext(Words)
End Sub

Private Sub TextBox1_KeyPress(ByVal sender As Object, ByVal e
As KeyPressEventArgs) Handles TextBox1.KeyPress
    If Not Char.IsLetter(e.KeyChar) And Not
Char.IsControl(e.KeyChar) Then
        e.KeyChar = ""
    End If
End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button1.Click
    Dim xlsApp As Object
    Dim xlsWb As Object
    Dim xlsSh As Object
    If TextBox1.Text = Label1.Text Then
        tn1 = 0
    End If
End Sub

```

```

tu1 = (ku1 - kd1).Milliseconds / 1000
tn2 = (kd2 - ku1).Milliseconds / 1000
tu2 = (ku2 - kd2).Milliseconds / 1000
tn3 = (kd3 - ku2).Milliseconds / 1000
tu3 = (ku3 - kd3).Milliseconds / 1000
tn4 = (kd4 - ku3).Milliseconds / 1000
tu4 = (ku4 - kd4).Milliseconds / 1000
tn5 = (kd5 - ku4).Milliseconds / 1000
tu5 = (ku5 - kd5).Milliseconds / 1000
tn6 = (kd6 - ku5).Milliseconds / 1000
tu6 = (ku6 - kd6).Milliseconds / 1000
tn7 = (kd7 - ku6).Milliseconds / 1000
tu7 = (ku7 - kd7).Milliseconds / 1000
tn8 = (kd8 - ku7).Milliseconds / 1000
tu8 = (ku8 - kd8).Milliseconds / 1000
tn9 = (kd9 - ku8).Milliseconds / 1000
tu9 = (ku9 - kd9).Milliseconds / 1000
tn10 = (kd10 - ku9).Milliseconds / 1000
tu10 = (ku10 - kd10).Milliseconds / 1000

If Not System.IO.Directory.Exists(Direct & "\Поточний
ФС") Then
    System.IO.Directory.CreateDirectory(Direct &
"\Поточний ФС")
    ElseIf (Not File.Exists(Direct & "\Поточний
ФС\ФС_now.txt")) Then
        Dim g
        g = File.Create(Direct & "\Поточний
ФС\ФС_now.txt")
        g.Close()
    End If
    If Not System.IO.Directory.Exists(Direct &
"\Результат") Then
        System.IO.Directory.CreateDirectory(Direct &
"\Результат")
        ElseIf (Not File.Exists(Direct &
"\Результат\Result.txt")) Then
            Dim g
            g = File.Create(Direct & "\Результат\Result.txt")
            g.Close()
        End If

    Dim writer As New StreamWriter(Direct & "\Поточний
ФС\ФС_now.txt", False, System.Text.Encoding.GetEncoding(1251))
    writer.WriteLine(tn1)
    writer.WriteLine(tu1)
    writer.WriteLine(tn2)
    writer.WriteLine(tu2)
    writer.WriteLine(tn3)
    writer.WriteLine(tu3)
    writer.WriteLine(tn4)
    writer.WriteLine(tu4)
    writer.WriteLine(tn5)

```



```

writer.WriteLine(tu5)
writer.WriteLine(tn6)
writer.WriteLine(tu6)
writer.WriteLine(tn7)
writer.WriteLine(tu7)
writer.WriteLine(tn8)
writer.WriteLine(tu8)
writer.WriteLine(tn9)
writer.WriteLine(tu9)
writer.WriteLine(tn10)
writer.WriteLine(tu10)
writer.Close()
Dim Txt As String = IO.File.ReadAllText(Direct &
"\Поточний ФС\ФС_now.txt", System.Text.Encoding.Default)
Txt = Txt.Replace(", ", ".")
IO.File.WriteAllText(Direct & "\Поточний
ФС\ФС_now.txt", Txt, System.Text.Encoding.Default)

Dim ТипМатЛаб =
Type.GetTypeFromProgID("Matlab.Application")
Dim МатЛаб = Activator.CreateInstance(ТипМатЛаб)
Dim now As String = "data = load ('" & Direct &
"\Поточний ФС\ФС_now.txt');"
Dim result As String = "dlmwrite('" & Direct &
"\Результат\Result.txt', a(1,1))"
МатЛаб.visible = False
If Labell1.Text = "адекватний" Then
Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\адекватний_train.txt');"
Dim Команди = {train + "net=newcD';" + "net =
newsom(P, [1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
End If
If Labell1.Text = "бібліотека" Then
Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\бібліотека_train.txt');"
Dim Команди = {train + "P = D';" + "net =
newsom(P, [1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
End If
If Labell1.Text = "відеопотік" Then
Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\відеопотік_train.txt');"
Dim Команди = {train + "P = D';" + "net =
newsom(P, [1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}

```

```

        Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
    End If
    If Label1.Text = "гермомішок" Then
        Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\гермомішок_train.txt');"
        Dim Команди = {train + "P = D';" + "net =
newsom(P,[1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
        Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
    End If
    If Label1.Text = "дендропарк" Then
        Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\дендропарк_train.txt');"
        Dim Команди = {train + "P = D';" + "net =
newsom(P,[1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
        Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
    End If
    If Label1.Text = "зубочистка" Then
        Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\зубочистка_train.txt');"
        Dim Команди = {train + "P = D';" + "net =
newsom(P,[1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
        Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
    End If
    If Label1.Text = "катастрофа" Then
        Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\катастрофа_train.txt');"
        Dim Команди = {train + "P = D';" + "net =
newsom(P,[1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
        Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
    End If
    If Label1.Text = "латифундія" Then
        Dim train As String = "D = load ('" & Direct &
"\Навчання НМ\латифундія_train.txt');"
        Dim Команди = {train + "P = D';" + "net =
newsom(P,[1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
        Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)

```

```

End If
If Labell1.Text = "менеджмент" Then
    Dim train As String = "D = load (' & Direct &
"\Навчання НМ\менеджмент_train.txt');"
    Dim Команди = {train + "P = D';" + "net =
newsom(P,[1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
    Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
End If
If Labell1.Text = "наповнення" Then
    Dim train As String = "D = load (' & Direct &
"\Навчання НМ\наповнення_train.txt');"
    Dim Команди = {train + "P = D';" + "net =
newsom(P,[1 2]);" + "net = train(net, P);" + now + "a = sim(net,
data);" + result}
    Dim Результат As String =
ТипМатЛаб.InvokeMember("Execute",
Reflection.BindingFlags.InvokeMethod, Nothing, МатЛаб, Команди)
End If

Dim Txt1 As String = IO.File.ReadAllText(Direct &
"\Результат\Result.txt", System.Text.Encoding.Default)
If Txt1 = 0 Then
    MessageBox.Show("ЗАДОВІЛЬНИЙ функціональний стан!
(*Бажаємо успіху в подальшій роботі*)", "Повідомлення!",
MessageBoxButtons.OK, MessageBoxIcon.Asterisk)
ElseIf Txt1 = 1 Then
    MessageBox.Show("НЕЗАДОВІЛЬНИЙ функціональний
стан! (*Відпочиньте та повертайтеся до роботи*)", "Повідомлення!",
MessageBoxButtons.OK, MessageBoxIcon.Stop)
End If

xlsApp = CreateObject("Excel.Application")
If (Not File.Exists(Direct &
"\Результат\Статистика.xls")) Then
    Dim g
    g = File.Create(Direct &
"\Результат\Статистика.xls")
    g.Close()
End If
xlsWb = xlsApp.WorkBooks.Open(Direct &
"\Результат\Статистика.xls")
xlsSh = xlsWb.Sheets(1)
Dim i = 2
Do
    If xlsSh.cells(i, 1).Value = "" Then
        xlsSh.cells(i, 1).Value = CStr(DateTime.Now)
        Exit Do
    End If
    i = i + 1
Loop

```

```

        i = 2
        Do
            If xlsSh.cells(i, 2).Value = "" Then
                If Txt1 = 0 Then
                    xlsSh.cells(i, 2).Value = "Задовільний
функціональний стан"
                ElseIf Txt1 = 1 Then
                    xlsSh.cells(i, 2).Value = "Незадовільний
функціональний стан"
                End If
            Exit Do
        End If
        i = i + 1
    Loop
    i = 2
    Do
        If xlsSh.cells(i, 3).Value = "" Then
            xlsSh.cells(i, 3).Value = Label1.Text
            Exit Do
        End If
        i = i + 1
    Loop
    xlsWb.Close(True)
    xlsApp.Quit()
    Else : MessageBox.Show("Будь ласка, перевірте правильність
введення даного слова!", "Помилка!", MessageBoxButtons.OK,
MessageBoxIcon.Warning)
    End If
    Label1.Text = GetNext(Words)
    TextBox1.Text = ""
    TextBox1.Select()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button2.Click
    If Dir(Direct & "\Результат\Статистика.xls") = "" Then
        MsgBox("Файлу не існує!", MessageBoxIcon.Warning)
    Else
        Process.Start(Direct & "\Результат\Статистика.xls")
    End If
End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button3.Click
    Try
        If Dir(Direct & "\Результат\Статистика.xls") = "" Then
            MsgBox("Файлу не існує!", MessageBoxIcon.Warning)
        Else
            File.Delete(Direct & "\Результат\Статистика.xls")
            MsgBox("Файл видалено!",
MessageBoxIcon.Information)
        End If
    Catch ex As Exception

```

```

        If ex.TargetSite.Name = "WinIOError" Then MsgBox("Файл
зайнятий іншим додатком!", MessageBoxIcon.Information)
    End Try
End Sub

Private Sub Button4_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button4.Click
    Me.Visible = False
    Form3.Visible = True
End Sub
End Class

```

Код для «Видалення файлів із вхідними даними» (Form5)

```

Imports System.IO

Public Class Form5

    Private Sub Form5_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        For Each foundFile As String In
My.Computer.FileSystem.GetFiles(Direct & "\Навчання НМ\")
            ComboBox1.Items.Add(foundFile)
        Next
    End Sub

    Private Sub someButton_MouseHover(ByVal sender As Object,
ByVal e As System.EventArgs) _
Handles Button1.MouseHover
        Dim toolTip As New ToolTip()
        toolTip.SetToolTip(Button1, "Видалити вибраний файл")
        toolTip.SetToolTip(Button2, "Видалити всі файли")
    End Sub

    Private Sub Form5_Closing(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.FormClosing
        Form1.Visible = True
    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button1.Click 'one
        If ComboBox1.SelectedIndex = -1 Then
            MsgBox("Оберіть файл з випадного списку!",
MessageBoxIcon.Warning)
        Else : File.Delete(ComboBox1.Text.ToString())
            ComboBox1.Items.Clear()
            For Each foundFile As String In
My.Computer.FileSystem.GetFiles(Direct & "\Навчання НМ\")
                ComboBox1.Items.Add(foundFile)
            Next
            ComboBox1.Text = ""
            MsgBox("Файл " & ComboBox1.Text.ToString() & "
видалено!", MessageBoxIcon.Information)
        End If
    End Sub

```

```

End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles Button2.Click 'all
    If
CreateObject("Scripting.FileSystemObject").FolderExists(Direct &
"\Навчання НМ") Then
        If
CreateObject("Scripting.FileSystemObject").GetFolder(Direct &
"\Навчання НМ").Files.Count > 0 Then
            For Each txt$ In IO.Directory.GetFiles(Direct &
"\Навчання НМ", "*.txt")
                IO.File.Delete(txt)
            Next
            ComboBox1.Items.Clear()
            For Each foundFile As String In
My.Computer.FileSystem.GetFiles(Direct & "\Навчання НМ\")
                ComboBox1.Items.Add(foundFile)
            Next
            MsgBox("Файли видалено!",
MessageBoxIcon.Information)
        Else : MsgBox("Папка " & Direct & "\Навчання НМ" & "
пуста!", MessageBoxIcon.Warning)
        End If
    Else : MsgBox("Папки " & Direct & "\Навчання НМ" & " не
існує!", MessageBoxIcon.Warning)
    End If
End Sub
End Class

```

Код для «Автор» (Form6)

```

Public Class Form6
    Private Sub Form6_Closed(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.FormClosed
        Form3.Visible = True
    End Sub
End Class

```

ДОДАТОК Б – Інструкція користувача

Для коректної роботи програми «Визначення ФС» необхідно виконати дві дії:

- 1. Встановити Matlab R2011b або вище.*
- 2. Встановити програму «Визначення ФС» за допомогою файлу «Setup_FS.msi».*

Установка інформаційної системи на комп'ютер

Якщо на комп'ютері оператора не встановлений Matlab R2011b або вище, тоді слід його встановити.

Далі необхідно зберегти пакет установника Windows (.msi) у зручному для користувача місці. Наприклад, як показано на рис. Б.1.

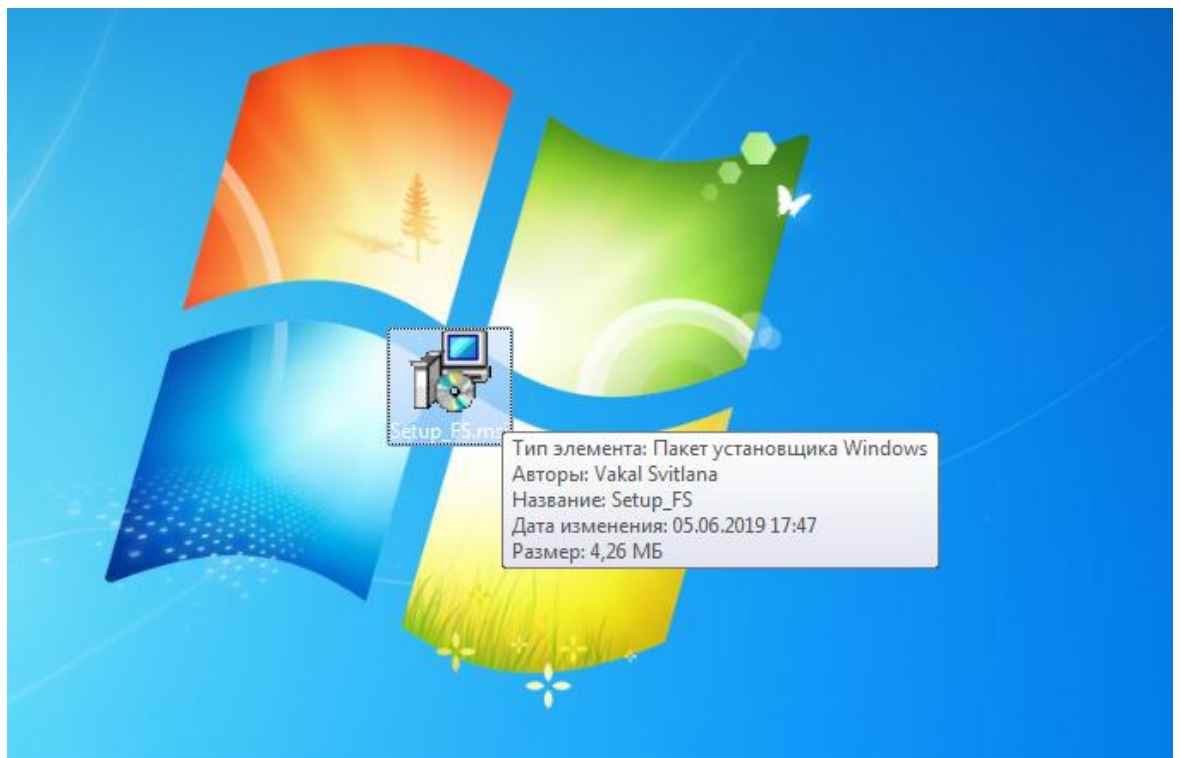


Рисунок Б.1 – Пакет установника знаходиться на Робочому столі

Натиснувши на нього двічі лівою клавшею мишки або вибравши команду у випадному вікні «Установити» встановити систему на комп'ютер. Після виконання даної дії відкривається стартове вікно встановлення програми для оцінки функціонального стану операторів (рис. Б.2).

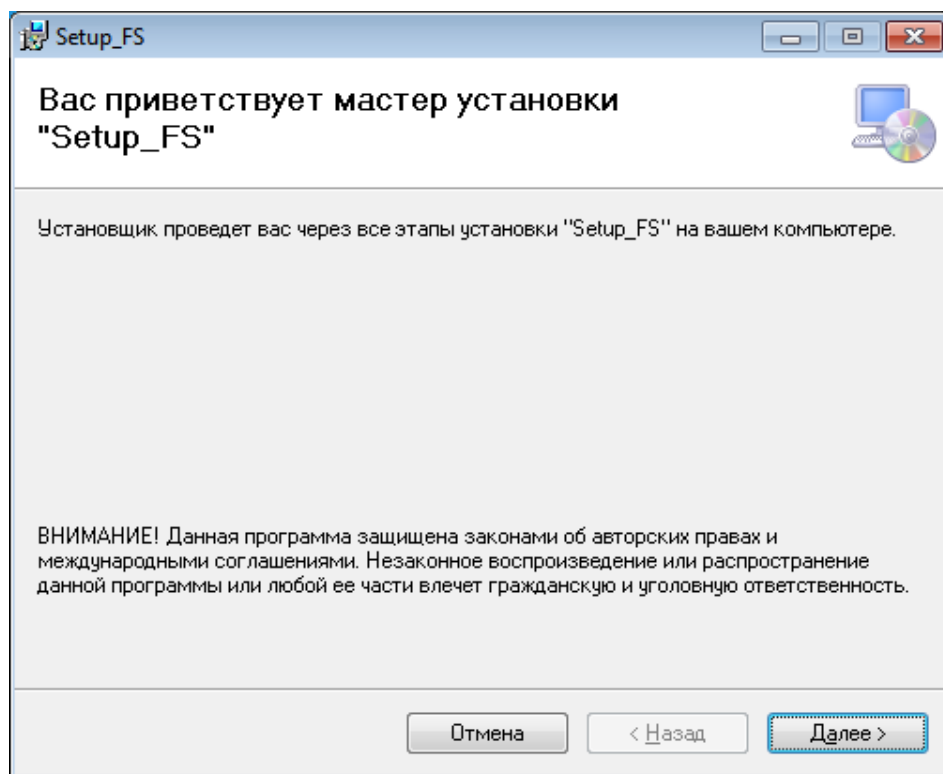


Рисунок Б.2 – Стартове вікно встановлення програми «Визначення ФС»

Натиснувши кнопку «Далее» користувач має змогу обрати до якої з директорій встановити інформаційну систему (рис. Б.3).

Обравши папку натиснути «Далее» та підтвердити установку програми на комп'ютер оператора (рис. Б.4).

Далі відбувається установка програми «Визначення ФС» (рис. Б.5) і по завершенню відкривається вікно з інформацією про успішне або неуспішне завершення роботи майстра установки (рис. Б.6).

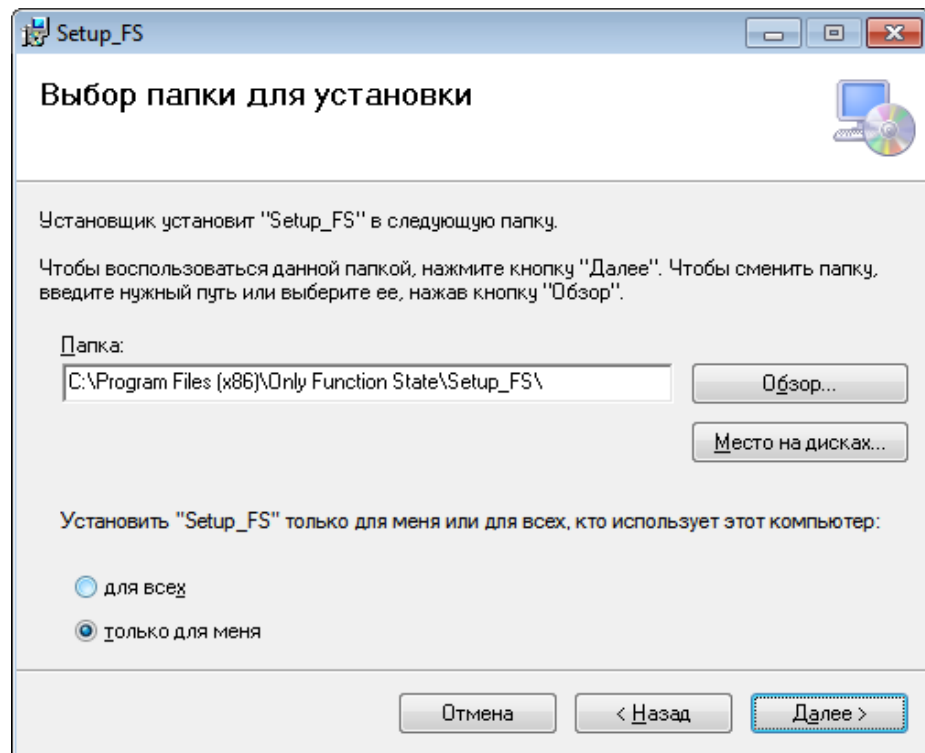


Рисунок Б.3 – Вибір директорії для встановлення програми «Визначення ФС»

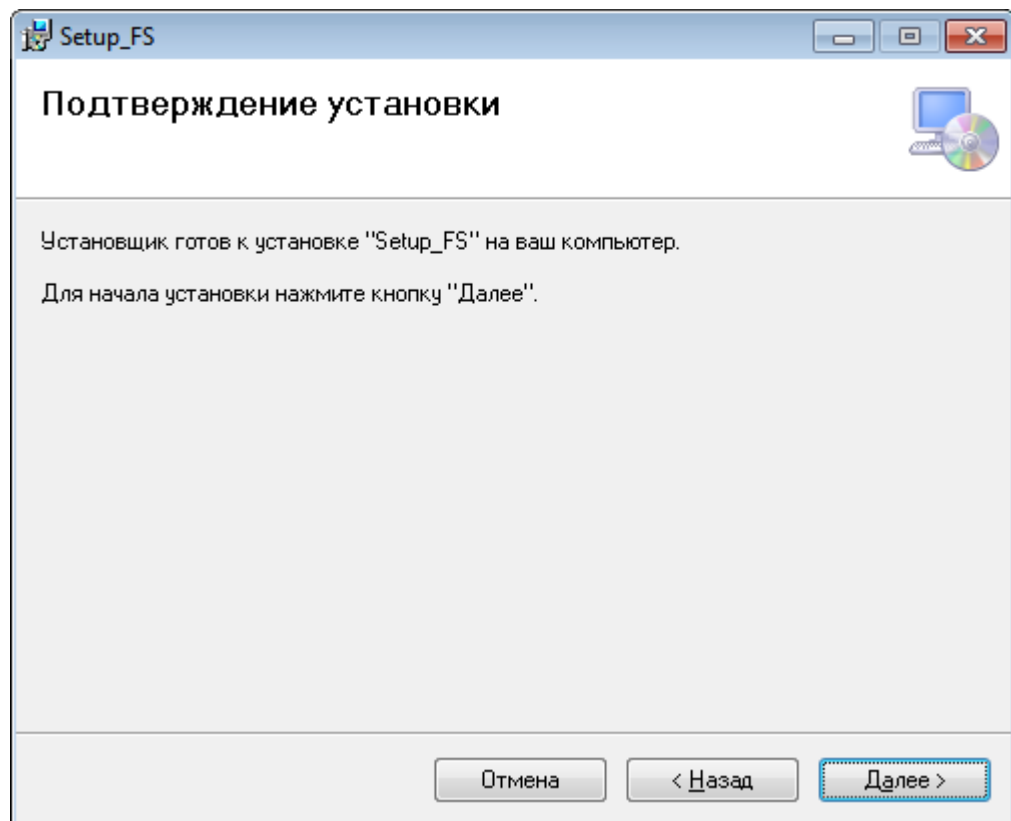


Рисунок Б.4 – Підтвердження встановлення програми «Визначення ФС»

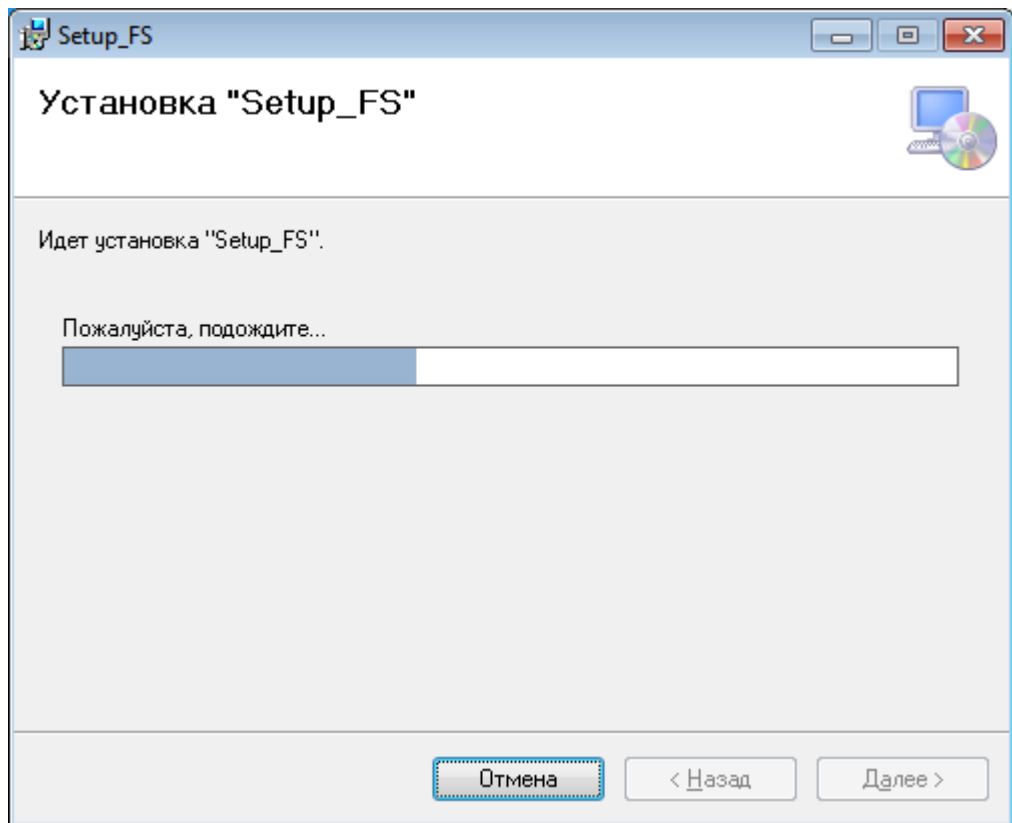


Рисунок Б.5 – Установка програми «Визначення ФС»

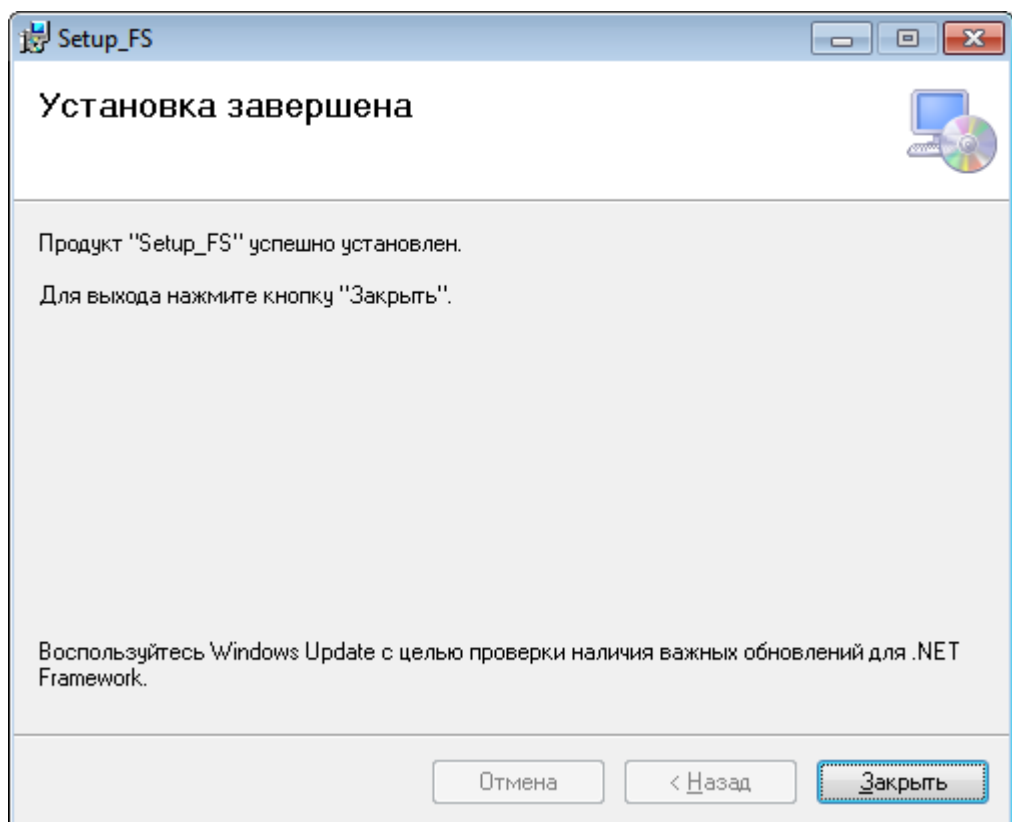


Рисунок Б.6 – Успішне завершення установки

У результаті установки на робочому столі з'явиться ярлик програми «Визначення ФС» (рис. Б.7).

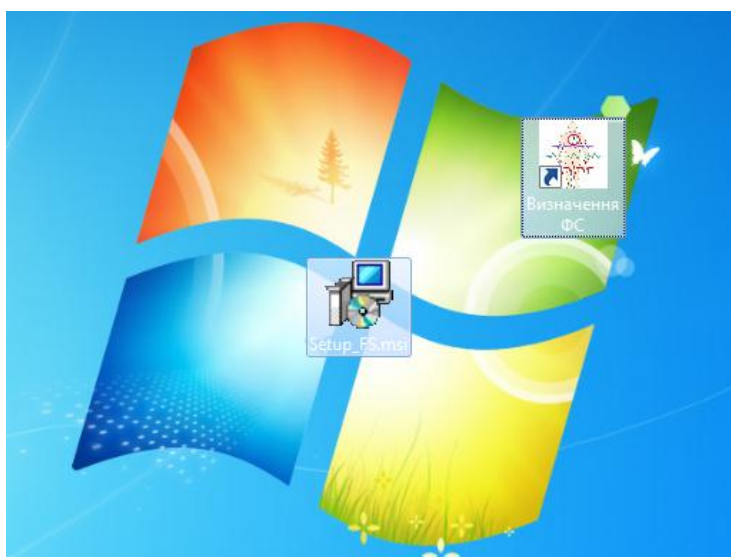


Рисунок Б.7 – Ярлик програми на робочому столі

Робота з програмою «Визначення ФС»

Запустити програму можна натиснувши двічі на ярлик програми «Визначення ФС». У результаті програму запущено, користувач бачить головне вікно програми (рис. Б.8).

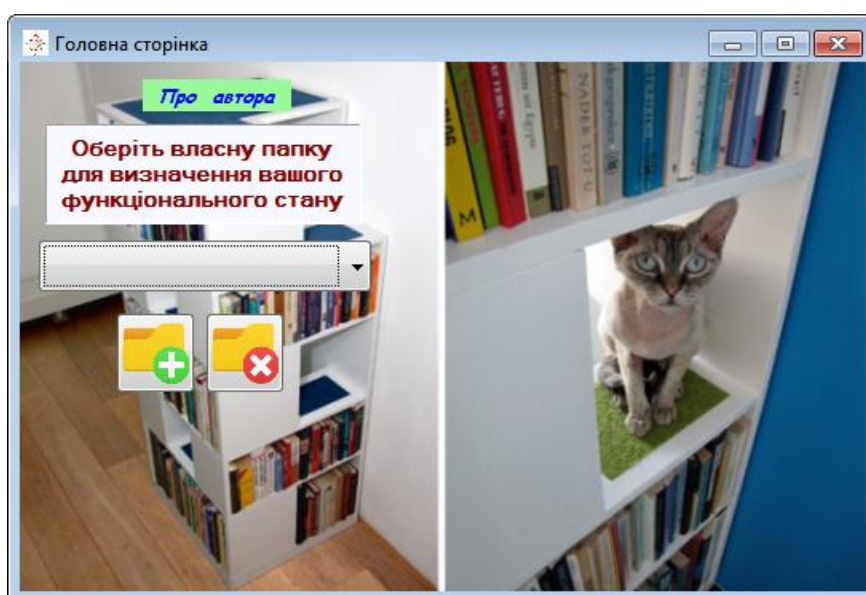


Рисунок Б.8 – Головне вікно програми

На вікні активні три кнопки: «Про автора», додавання та видалення папки. При натисканні відбуваються наступні дії, які показані на рис. Б.9 – Б.11.



Рисунок Б.9 – Вікно «Автор», після натискання «Про автора» на головному вікні програми

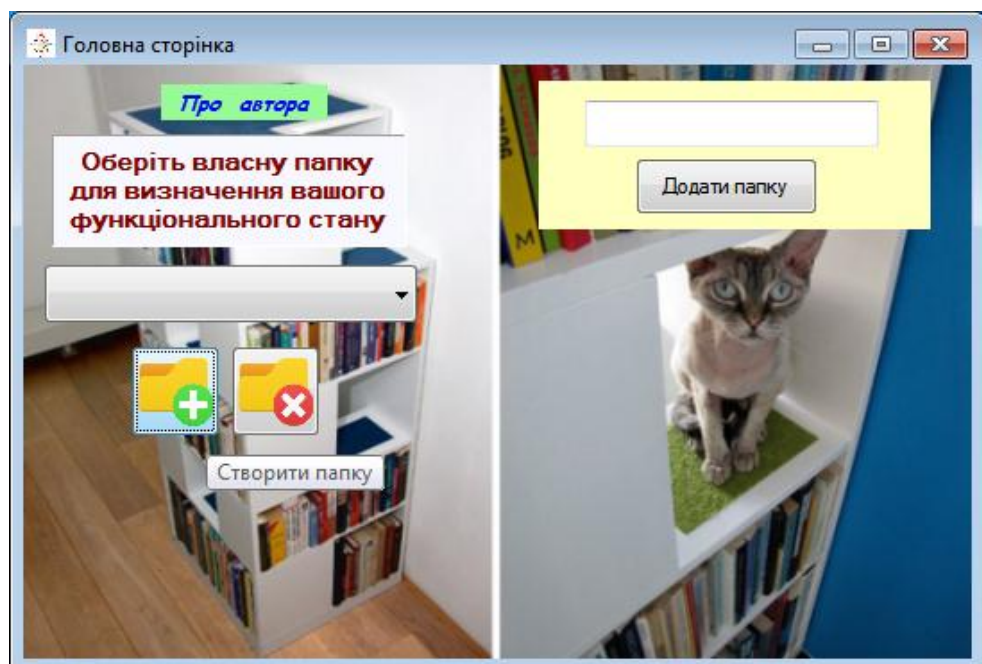


Рисунок Б.10 – Активна можливість створення папки

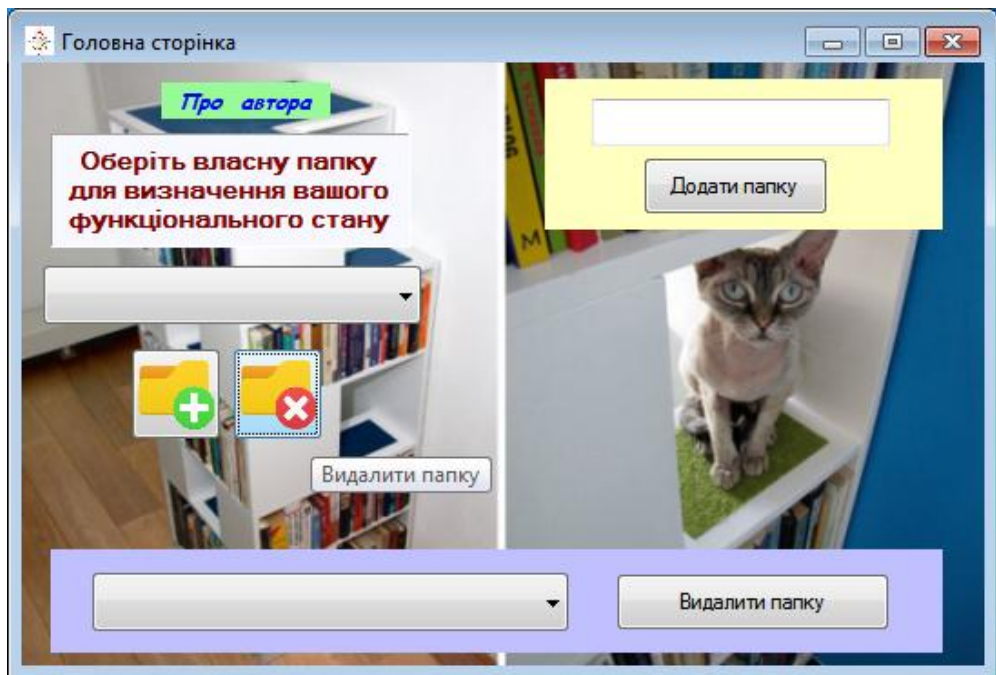


Рисунок Б.11 – Активна можливість видалення папки

Обираючи папку (рис. Г.12) користувач входить до директорії власної папки та має змогу виконати 2 дії (рис. Б.13), в залежності від того чи створив він навчальну вибірку чи ні.

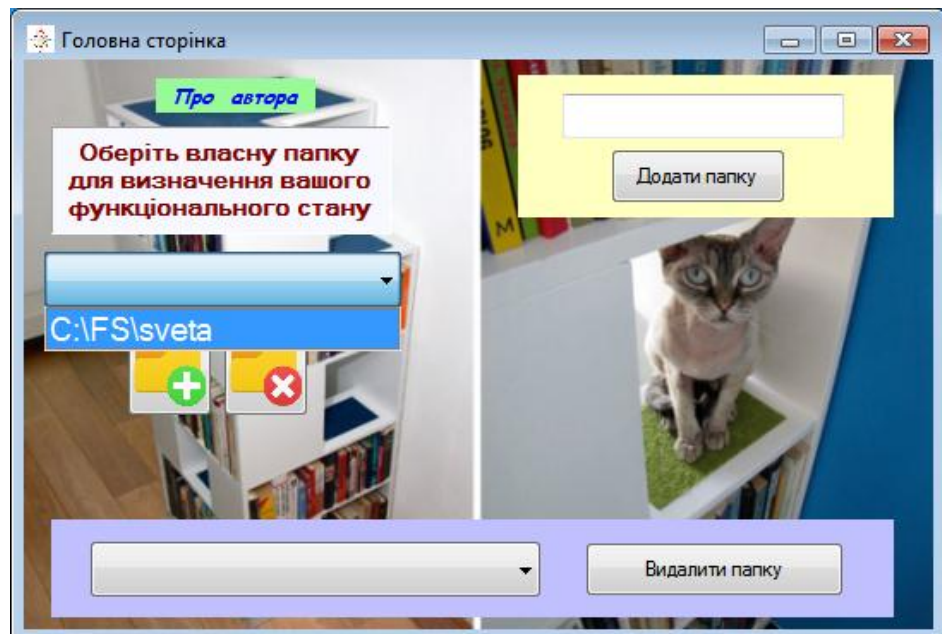


Рисунок Б.12 – Вибір папки

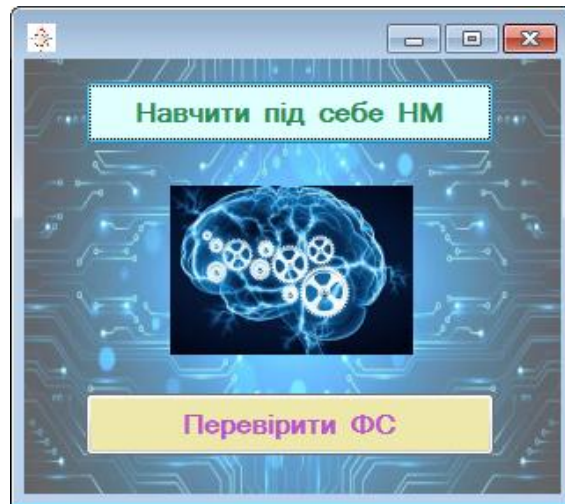


Рисунок Б.13 – Проміжне вікно вибору дії

Натиснувши на «Навчити під себе НМ» користувач опиняється на вікні «Персональні налаштування системи» (рис. Б.14). У цьому вікні відбувається створення навчальної вибірки.

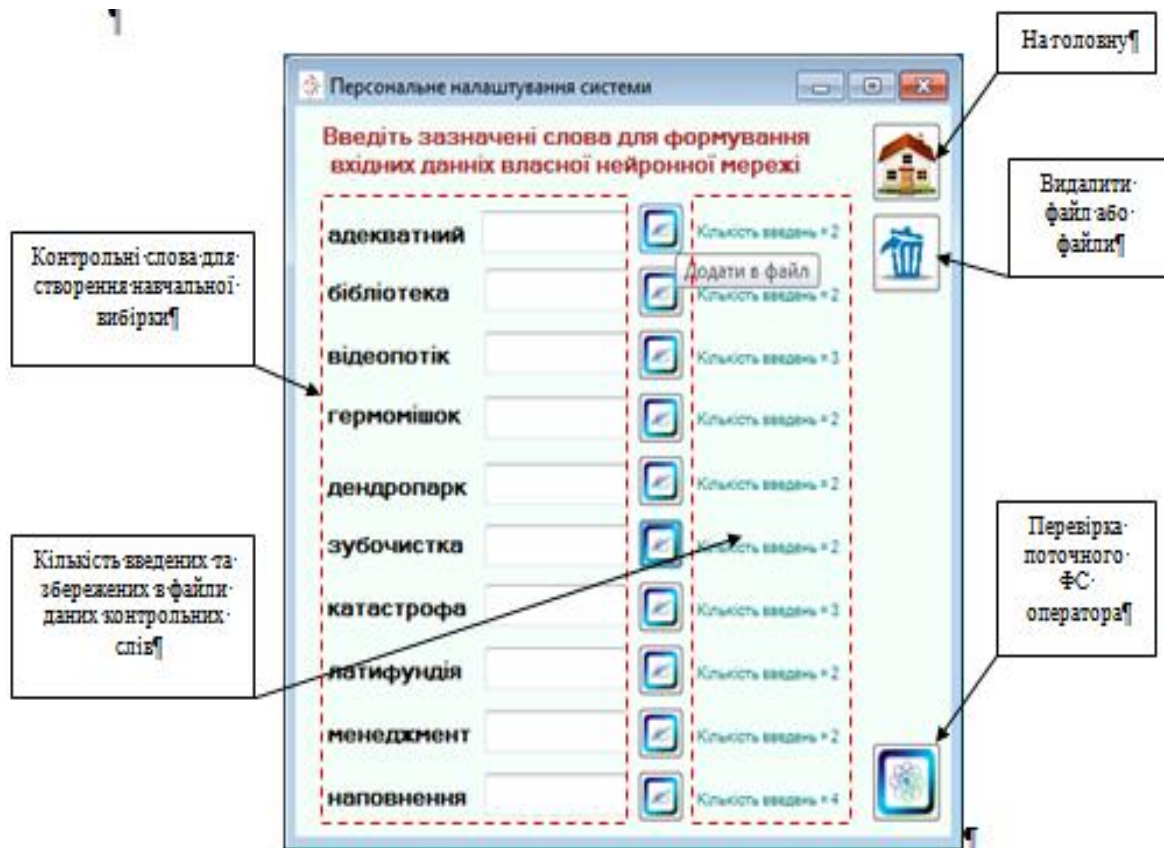


Рисунок Б.14 – Вікно «Персональні налаштування системи»

!!! Користувач обов'язково має заповнити файли для навчання нейронної мережі хоча б один раз, інакше НЕ зможе перевіряти власний функціональний стан. Для підказки використана технологія інформаційних повідомлень або повідомлень про помилки.

Відкривши вікно видалення файлів (рис. Б.15) користувач має обрати, який саме файл він бажає оновити, у випадку якщо це один файл, якщо всі то натиснути на кнопку видалення всіх файлів.

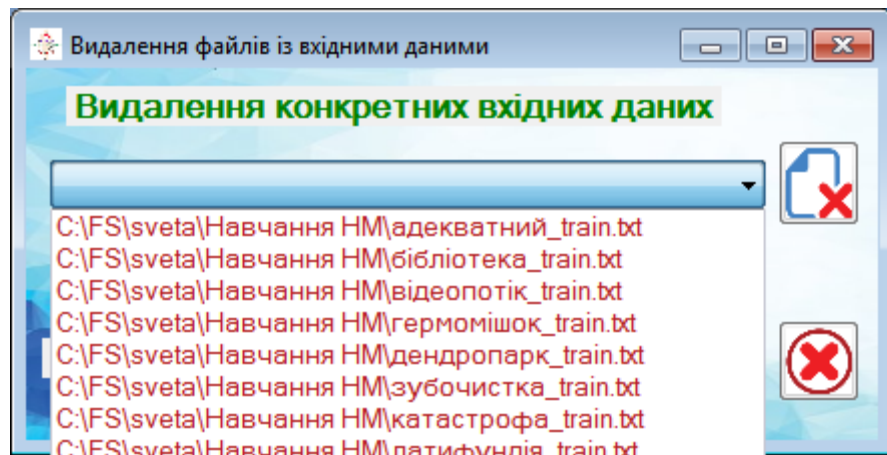


Рисунок Б.15 – Вікно видалення файлів

Натиснувши на «Перевірити ФС» користувач опиняється на вікні «Перевірка поточного ФС» (рис. Б.16). На цьому етапі роботи програми відбувається створення, навчання та симуляція Самоорганізованої мапи Кохонена, за якою визначається поточний стан оператора інформаційних систем. У результаті оператор отримує результат роботи мережі в текстовому файлі та у вигляді повідомлення (рис. Б.17).

Також після проходження перевірки можна перевірити статистичні дані, натиснувши кнопку «Перегляд статистики». У результаті відкриється Excel-файл (рис. Б.18) із такою інформацією як час проходження перевірки, її результат та за яким із слів вона відбувалася.

!!! Файл «Статистика» доступний лише для читання, тому вносити якісь нові дані неможливо.

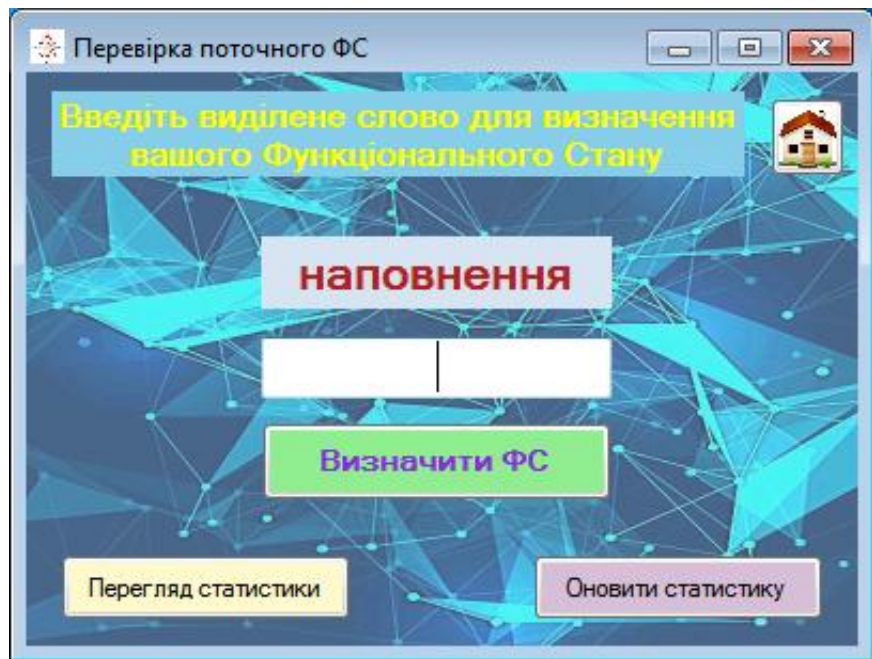


Рисунок Б.16 – Вікно «Перевірка поточного ФС»

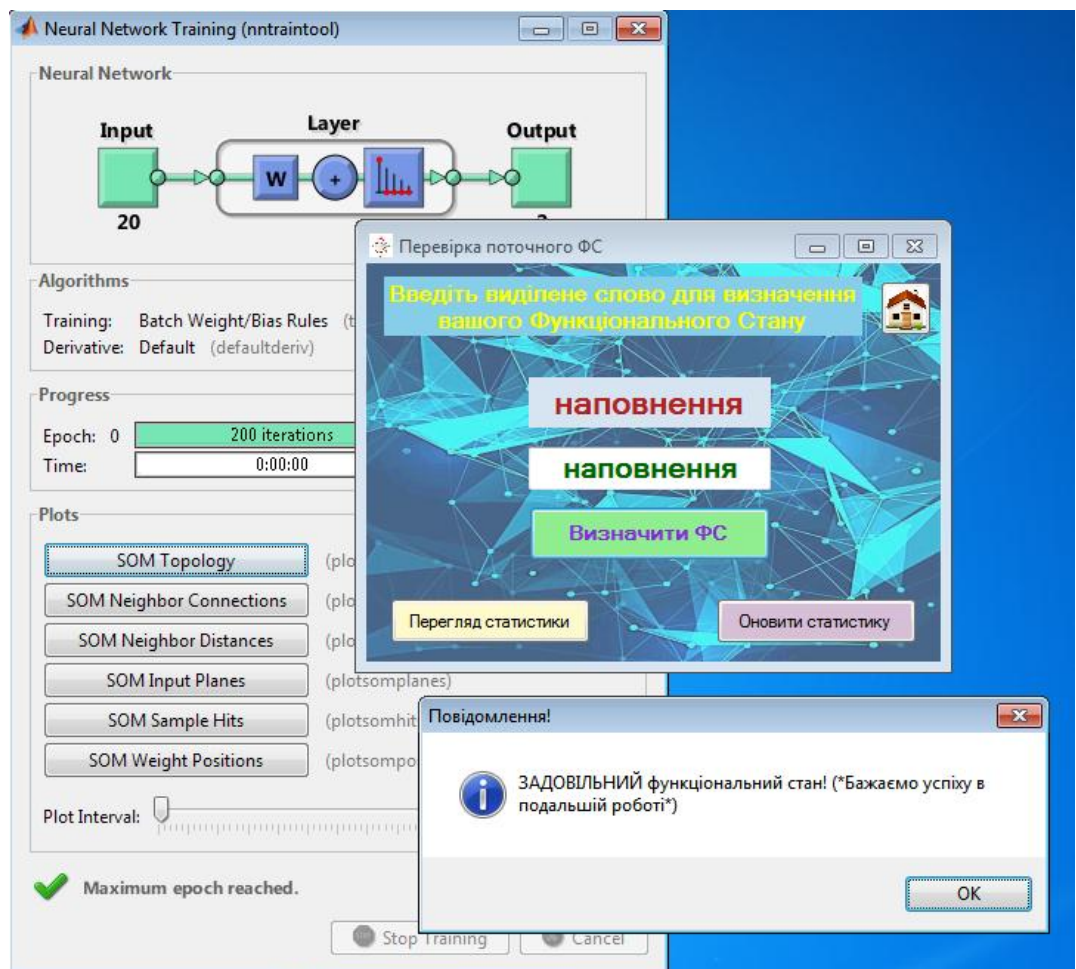


Рисунок Б.17 – Результат оцінки функціонального стану оператора

	A	B	C	D	E	F	G	H
1	05.06.2019 18:16	Задовільний функціональний стан	катастрофа					
2	05.06.2019 18:17	Незадовільний функціональний ста	дендропарк					
3	05.06.2019 18:18	Задовільний функціональний стан	зубочистка					
4	05.06.2019 18:58	Незадовільний функціональний ста	гермомішок					
5	05.06.2019 18:58	Задовільний функціональний стан	катастрофа					
6	05.06.2019 18:58	Незадовільний функціональний ста	менеджмент					
7	05.06.2019 18:58	Задовільний функціональний стан	гермомішок					
8	05.06.2019 18:59	Незадовільний функціональний ста	дендропарк					
9	05.06.2019 18:59	Задовільний функціональний стан	менеджмент					
10	05.06.2019 18:59	Незадовільний функціональний ста	бібліотека					
11	05.06.2019 18:59	Задовільний функціональний стан	дендропарк					
12	05.06.2019 19:00	Задовільний функціональний стан	гермомішок					
13	05.06.2019 19:00	Задовільний функціональний стан	зубочистка					
14	05.06.2019 19:03	Задовільний функціональний стан	адекватний					
15	05.06.2019 19:03	Задовільний функціональний стан	латифундія					
16	05.06.2019 19:04	Задовільний функціональний стан	наповнення					
17	05.06.2019 19:04	Задовільний функціональний стан	адекватний					
18	05.06.2019 19:04	Незадовільний функціональний ста	бібліотека					
19	05.06.2019 19:04	Задовільний функціональний стан	відеопотік					
20	05.06.2019 19:05	Задовільний функціональний стан	латифундія					
21	05.06.2019 19:05	Незадовільний функціональний ста	менеджмент					
22	05.06.2019 19:05	Незадовільний функціональний ста	наповнення					
23	05.06.2019 19:05	Задовільний функціональний стан	гермомішок					
24	05.06.2019 19:06	Задовільний функціональний стан	дендропарк					
25	05.06.2019 19:06	Незадовільний функціональний ста	гермомішок					
26	05.06.2019 19:06	Незадовільний функціональний ста	латифундія					
27	05.06.2019 19:06	Задовільний функціональний стан	гермомішок					
28	05.06.2019 19:07	Незадовільний функціональний ста	бібліотека					
29	07.06.2019 9:17	Задовільний функціональний стан	наповнення					
30								

Рисунок Б.18 – Вміст файлу «Статистика» одного з операторів

За допомогою кнопки «Оновити статистику» відбувається видалення файлу зі статистичними даними (рис. Б.19). Для створення нового слід пройти перевірку поточного функціонального стану, адже створення файлу автоматичне.

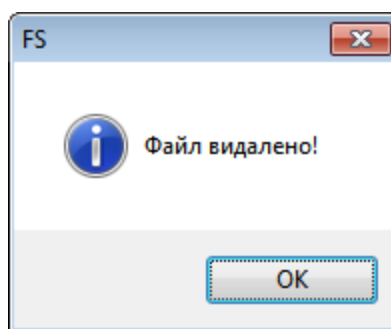


Рисунок Б.19 – Повідомлення про видалення файлу «Статистика»

Після кожного з переходів до нових вікон користувач має змогу повернутися на попереднє вікно за допомогою закриття поточного вікна.

У програмі передбачено велику кількість інформаційних повідомлень та повідомлень про помилки. Деякі з них наведені на рис. Б.20.

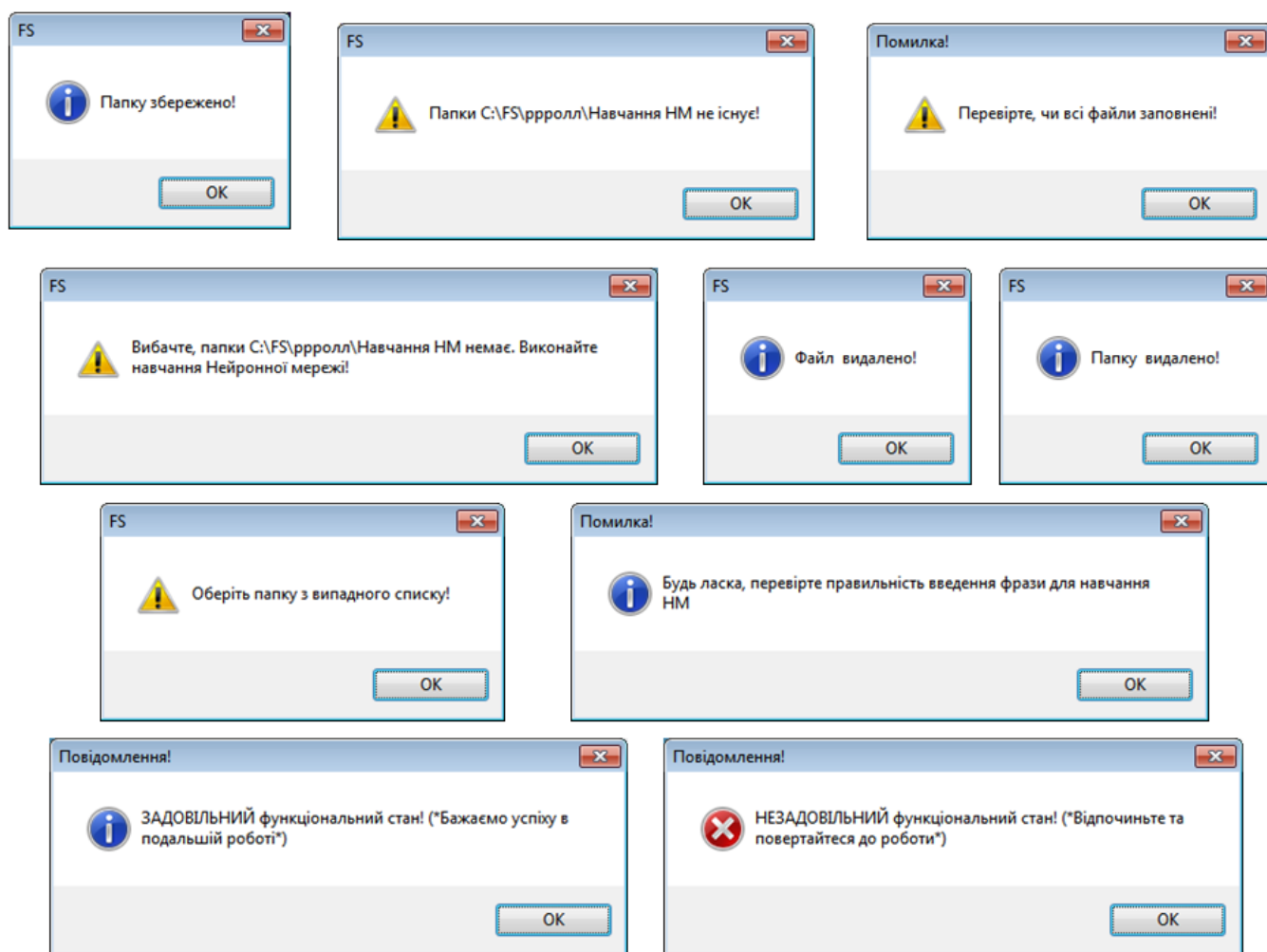


Рисунок Б.20 – Приклади деяких повідомлень, які видає програма «Визначення ФС»

Видалення інформаційної системи з комп'ютера

Один із найпростіших методів видалення програми – видалення за допомогою пакету установки встановленої програми. Для цього необхідно натиснути двічі лівою кlawішею мишки на ярлик установки програми «Визначення ФС» – «Setup_FS». У результаті користувачу буде запропоновано або «Восстановить “Setup_FS”» або «Удалить “Setup_FS”» (рис. Б.21).

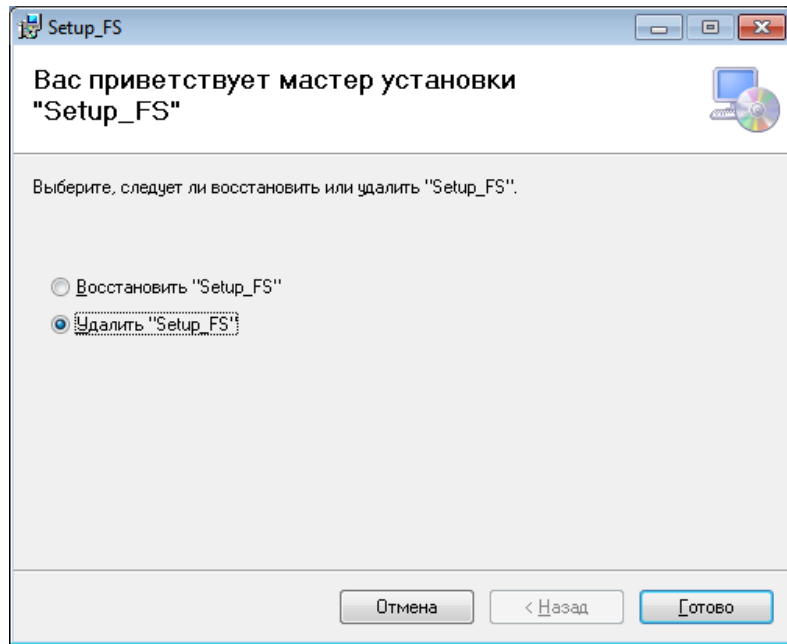


Рисунок Б.21 – Видалення програми «Визначення ФС» за допомогою майстра установки

Слід обрати «Удалить “Setup_FS”» для того, щоб програма «Визначення ФС» видалилася з комп’ютера оператора. У результаті натискання клавіші «Готово», вікно майстра установки видасть результат видалення (рис. Б.22).

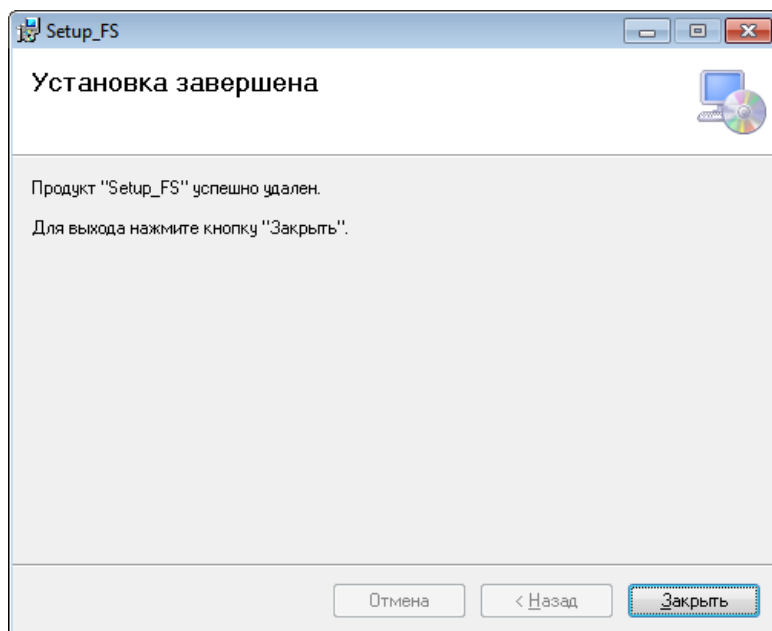


Рисунок Б.22 – Систему оцінки функціонального стану оператора видалено

ДОДАТОК В – Експериментальні дослідження

Для підтвердження точності роботи програми оцінки функціонального стану людини-оператора інформаційних систем проведено ряд експериментальних досліджень.

Експерименти проводилися впродовж одного робочого дня за допомогою 5-ти людей-операторів, які проходили тестування власного функціонального стану оцінюючи його двома критеріями.

1. Оцінка за думкою користувача програми власного функціонального стану.
2. Оцінка, яку отримали після проходження перевірки програмою «Визначення ФС».

Оператори мали пройти перевірку кожної робочої години. Тому в результаті вийшло, що у кожного з операторів результатів мало бути 10 власноруч відміченими та 10 визначеними машиною, що записувалися до файлу «Статистика». Для зручності подальшого дослідження експериментів користувачі стави «1», якщо психофізичний стан задовільний, та «0» - незадовільний.

Для побудову графіків точності програмного продукту було створено таблицю в MS Excel з даними про щогодинний поточний стан 5-ти операторів (рис. В.1).

	A	B	C	D	E	F	G	H	I	J	K
1	Години	Оператор 1	Оператор 2	Оператор 3	Оператор 4	Оператор 5					
2	8:00	0	0	1	1	1	1	1	0	1	1
3	9:00	1	1	1	1	1	0	1	1	1	1
4	10:00	1	0	1	1	1	0	1	1	0	1
5	11:00	1	1	1	1	1	1	0	1	1	1
6	12:00	0	0	0	0	1	1	0	0	1	1
7	13:00	1	1	1	1	1	1	1	1	1	1
8	14:00	1	0	1	1	1	1	1	1	1	1
9	15:00	1	1	1	0	1	0	0	0	1	1
10	16:00	1	1	1	1	0	0	0	0	1	1
11	17:00	0	1	1	1	0	0	1	1	0	0

Рисунок В.1 – Зображення таблиці порівнянь людських відчуттів і машинного розуму

У таблиці позначено 5 операторів, що проходили перевірку, та їх результати по власним відчуттям (білий колір стовпця) і по математичній моделі програмного продукту (зелений колір).

За результатами дослідження по кожному з операторів побудовано графіки відповідності людських відчуттів і машинного розуму (рис. В.2 – В.6).

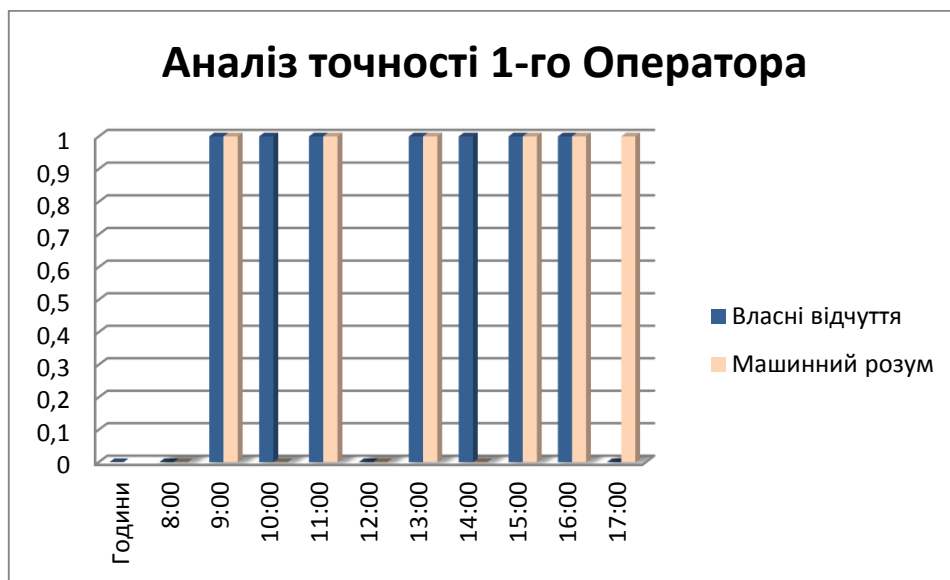


Рисунок В.2 – Результат порівнянь першого оператора



Рисунок В.3 – Результат порівнянь другого оператора



Рисунок В.4 – Результат порівнянь третього оператора



Рисунок В.5 – Результат порівнянь четвертого оператора



Рисунок В.6 – Результат порівнянь п'ятого оператора

Розрахуємо середнє значення точності методу для кожного з операторів окремо.

$$Correctness = \frac{\sum_{i=1}^n \sum_{j=1}^s \alpha_{ij}}{n \cdot s}, \quad (B.1)$$

де i – номер експерименту, $i = 1, n$; (n – кількість експериментів);

j – номер людини оператора, $j = 1, s$; (s – кількість операторів);

$\alpha_{ij} = \begin{cases} 1, & \text{якщо оцінка програми співпадає з оцінкою оператора} \\ 0, & \text{якщо оцінка програми НЕ співпадає з оцінкою оператора} \end{cases}$

Середнє значення точності для першого оператора = 0,7.

Середнє значення точності для другого оператора = 0,9.

Середнє значення точності для третього оператора = 0,7.

Середнє значення точності для четвертого оператора = 0,8.

Середнє значення точності для п'ятого оператора = 0,9.

Узагальнений графік характеристики точності зображений на рис. В.7.

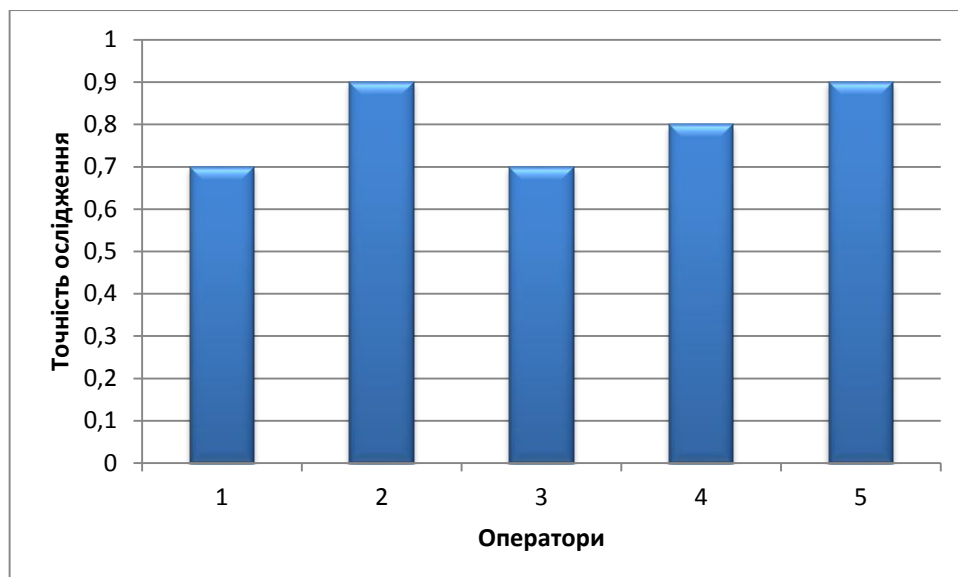


Рисунок В.7 – Узагальнена характеристика точності

Для розрахунку середнього значення точності методу для програмного продукту «Визначення ФС» слід зібрати до купи всі перевірки точності операторів та загальну суму експериментів.

Проведений дослід показує, що середнє значення точності методу рівне 0,8.

Отже, експериментальні дані підтверджують математичну модель для оцінки функціонального стану операторів інформаційних систем.