

Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
« ____ » _____ 2023 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація хмарної системи
стискування та перетворення форматів зображень”

Виконав здобувач вищої освіти
II курсу, групи КН-22М-1
ОПП «Комп’ютерні науки»
спеціальності 122 «Комп’ютерні науки»
_____ Бабаєв Р.І.
« ____ » _____ 2023 р.

Керівник проекту
доктор технічних наук, професор
_____ Смірнов О.А.
« ____ » _____ 2023 р.
Рецензент _____

Центральноукраїнський національний технічний університет
Факультет *Механіко-технологічний*
Кафедра *Кібербезпеки та програмного забезпечення*
Рівень вищої освіти *магістр*
Галузь знань *12* "Інформаційні технології"
Спеціальність *122* "Комп'ютерні науки"
Освітньо-професійна (освітньо-наукова) програма "Комп'ютерні науки"

ЗАТВЕРДЖУЮ

Завідувач кафедри

д.т.н., проф.

Олексій СМІРНОВ

« 6 » вересня 2023 року

ЗАВДАННЯ НА ВИПУСКНУ КВАЛІФІКАЦІЙНУ РОБОТУ ЗА ДРУГИМ (МАГІСТЕРСЬКИМ) РІВНЕМ ВИЩОЇ ОСВІТИ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

Бабасву Роману Ілгаровичу

(прізвище, ім'я, по батькові)

- | | | | | | | | | | | | |
|--|--|--|----------------------------|---|---|--|--|--|---------------------|--|--|
| 1. Тема роботи | <i>Дослідження та програмна реалізація хмарної системи стискування та перетворення форматів зображень</i> | | | | | | | | | | |
| 2. Керівник роботи | <i>Смірнов Олексій Анатолійович, докт. техн. наук, професор</i>
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання) | | | | | | | | | | |
| затверджені наказом вищого навчального закладу № 32-13 від 04.08.2023 року | | | | | | | | | | | |
| 3. Строк подання студентом роботи до захисту | <i>10.12.2023 р.</i> | | | | | | | | | | |
| 4. Мета та завдання випускної кваліфікаційної роботи: | <i>Метою розробки є дослідження та програмна реалізація хмарної системи стискування та перетворення форматів зображень</i> | | | | | | | | | | |
| 5. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) | <table border="0"><tr><td><i>1. Призначення та область використання.</i></td><td><i>6. Наукова новизна.</i></td></tr><tr><td><i>2. Перегляд аналогічних існуючих систем.</i></td><td><i>7. Економічна ефективність розробленої програми.</i></td></tr><tr><td><i>3. Опис і обґрунтування проектних рішень.</i></td><td><i>8. Заходи з охорони праці та техніки безпеки.</i></td></tr><tr><td><i>4. Етапи програмування системи.</i></td><td><i>9. Висновки.</i></td></tr><tr><td><i>5. Впровадження системи в промислову експлуатацію</i></td><td></td></tr></table> | <i>1. Призначення та область використання.</i> | <i>6. Наукова новизна.</i> | <i>2. Перегляд аналогічних існуючих систем.</i> | <i>7. Економічна ефективність розробленої програми.</i> | <i>3. Опис і обґрунтування проектних рішень.</i> | <i>8. Заходи з охорони праці та техніки безпеки.</i> | <i>4. Етапи програмування системи.</i> | <i>9. Висновки.</i> | <i>5. Впровадження системи в промислову експлуатацію</i> | |
| <i>1. Призначення та область використання.</i> | <i>6. Наукова новизна.</i> | | | | | | | | | | |
| <i>2. Перегляд аналогічних існуючих систем.</i> | <i>7. Економічна ефективність розробленої програми.</i> | | | | | | | | | | |
| <i>3. Опис і обґрунтування проектних рішень.</i> | <i>8. Заходи з охорони праці та техніки безпеки.</i> | | | | | | | | | | |
| <i>4. Етапи програмування системи.</i> | <i>9. Висновки.</i> | | | | | | | | | | |
| <i>5. Впровадження системи в промислову експлуатацію</i> | | | | | | | | | | | |
| 6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) | | | | | | | | | | | |
| <i>Наукова новизна</i> | <i>1 аркуш</i> | | | | | | | | | | |
| <i>Структурна схема системи</i> | <i>1 аркуш</i> | | | | | | | | | | |
| <i>Функціональна схема системи</i> | <i>1 аркуш</i> | | | | | | | | | | |
| <i>Діаграма процесів</i> | <i>1 аркуш</i> | | | | | | | | | | |
| <i>Блок-схема алгоритму роботи додатку</i> | <i>2 аркуша</i> | | | | | | | | | | |
| <i>Показники економічної ефективності</i> | <i>1 аркуш</i> | | | | | | | | | | |

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічний	Савеленко Г.В.	05.10.2023	14.11.2023
Охорона праці	Оришака О.В.	06.10.2023	16.11.2023

7. Дата видачі завдання « 6 » вересня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Строк виконання етапів випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти	Примітка
1.	Аналіз існуючих систем	10.10.2023 р.	
2.	Постановка задачі, оформлення ТЗ	15.10.2023 р.	
3.	Розробка моделі компонента	20.10.2023 р.	
4.	Розробка структур даних	25.10.2023 р.	
5.	Розробка алгоритмів зв'язку та відображення	30.10.2023 р.	
6.	Програмування алгоритмів	10.11.2023 р.	
7.	Розрахунок економічної ефективності	13.11.2023 р.	
8.	Розрахунки з охорони праці та техніки безпеки	15.11.2023 р.	
9.	Оформлення ПЗ	17.11.2023 р.	
10.	Попередній захист роботи	10.12.2023 р.	

Дата видачі завдання
« 6 » вересня 2023 р.

Підпис керівника

(прізвище та ініціали)Завдання прийнято до виконання
« 6 » вересня 2023 р.

Підпис здобувача

(прізвище та ініціали)

АНОТАЦІЯ

Бабаєв Р.І. Дослідження та програмна реалізація хмарної системи стискування та перетворення форматів зображень. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2023.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для хмарної системи стискування та перетворення форматів зображень.

Метою розробки є дослідження та програмна реалізація хмарної системи стискування та перетворення форматів зображень.

Об'єктом дослідження є процес стискування та перетворення форматів зображень.

Предметом дослідження є методи стискування та перетворення форматів зображень.

Методи дослідження базуються на методах комп'ютерної графіки, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація хмарної системи стискування та перетворення форматів зображень.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ архітектури IBM PC з ОС Windows 10/11.

Програму розроблено в середовищі Delphi 10.4 Sydney.

Ключові слова: комп'ютерні науки, стискування та перетворення форматів зображень

ABSTRACT

Babaiev R.I. Research and software implementation of a cloud system for compression and conversion of image formats. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2023.

In this graduation thesis for the second (master's) level of higher education, software is developed, which is intended for a cloud-based system of compression and conversion of image formats.

The purpose of the development is research and software implementation of a cloud system for compression and conversion of image formats.

The object of research is the process of compression and conversion of image formats.

The subject of research is methods of compression and conversion of image formats.

Research methods are based on computer graphics methods, mathematical statistics methods, and software development methods.

The result of the work is a software implementation of a cloud system for compression and conversion of image formats.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

A convenient user interface has been developed. Instructions for working with software tools are provided.

The program can be used on PCs of IBM PC architecture with Windows 10/11 OS.

The program was developed in the Delphi 10.4 Sydney environment.

Keywords: computer science, compression and conversion of image formats

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ	3
ВСТУП.....	4
1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ	6
1.1 Призначення системи.....	6
1.2 Область застосування.....	7
2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ	9
2.1 Огляд існуючих систем, технологій, архітектур та програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	9
2.2 Обґрунтування вибору засобів для побудови системи та мови програмування.....	16
2.3 Розгорнута постановка завдання	21
3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ	23
3.1 Опис функціонування системи	23
3.2 Розробка структурної схеми.....	39
3.3 Розробка функціональної схеми	41
3.4 Розробка діаграми процесів.....	45
4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ВІРНІСТЬ ПРОЕКТНИХ ТА ПРОГРАМНИХ РІШЕНЬ.....	47
4.1 Розробка блок-схем та опис алгоритмів функціонування системи.....	47
4.2 Захист розробленого програмного забезпечення.....	58
5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ	62
6 НАУКОВА НОВИЗНА	68

						ВКРМ-122.23.0027.00.00.ПЗ		
Вим	Арк.	№ докум.	Підп.	Дата		Літ.	Аркуш	Аркушів
Розроб.	Бабаєв Р.І.				Дослідження та програмна реалізація хмарної системи стискання та перетворення форматів зображень	М	1	108
Перев.	Смірнов О.А.					ЦНТУ КН-22М-1		
Н.контр.	Коваленко А.С.							
Затв.	Смірнов О.А.							

7 ЕКОНОМІЧНА ЕФЕКТИВНІСТЬ РОЗРОБЛЕНОЇ ПРОГРАМИ.....	69
7.1 Техніко економічне обґрунтування теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.....	69
7.2 Розрахунок трудомісткості розробки програмної продукції.....	71
7.3 Визначення чисельності виконавців і планового фонду зарплати.....	73
7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника.....	77
7.5 Визначення собівартості розробки та ціни програмної продукції.....	82
7.6 Визначення об'єму капітальних вкладень та експлуатаційних витрат у споживача програмної продукції.....	85
7.7 Визначення експлуатаційних витрат.....	85
7.8 Визначення економічної ефективності програмної продукції.....	87
7.9 Висновок.....	89
8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ	90
8.1 Вступ.....	90
8.2 Шкідливі і небезпечні фактори при роботі з комп'ютером.....	91
8.3 Пожежна безпека.....	92
8.4 Розробка заходів з умов поліпшення охорони праці.....	94
8.5 Розрахункова частина	95
9 ОСНОВНІ ВИСНОВКИ.....	100
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	102

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ДКП	–	дискретне косинусоїдальне перетворення
МК	–	матриця квантування
BMP	–	Bitmap Picture – графічний формат
DDB	–	апаратно-залежні растри
DIB	–	апаратно-незалежні растри
GIF	–	Graphics Interchange Format – графічний формат
ISO	–	міжнародна організація по стандартизації
JPEG	–	Joint Photographic Experts Group – графічний формат
LZW	–	алгоритм Лемпеля – Зіва – Велча стискання інформації
RLE	–	Run Length Encoding – групове кодування

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		3

ВСТУП

Актуальність теми. Протягом останніх 10 років у рамках комп'ютерної графіки бурхливо розвивається зовсім нова область – алгоритми архівації зображень. Поява цієї області обумовлене тим, що зображення – це своєрідний тип даних, характеризуємий трьома особливостями. Зображення (як і відео) займають набагато більше місця в пам'яті, ніж текст. Ця особливість зображень визначає актуальність алгоритмів архівації графіки. Другою особливістю зображень є те, що людський зір при аналізі зображення оперує контурами, загальним переходом кольорів і порівняно невідчутно до малих змін у зображенні. Таким чином, ми можемо створити ефективні алгоритми архівації зображень, у яких декомпресоване зображення не буде збігатися з оригіналом, однак людина цього не помітить. Дана особливість людського зору дозволила створити спеціальні алгоритми стиску, орієнтовані тільки на зображення. Ці алгоритми мають дуже високі характеристики. Ми можемо легко помітити, що зображення у відмінність, наприклад, від тексту має надмірність в 2-х вимірах. Тобто, як правило, сусідні крапки, як по горизонталі, так і по вертикалі в зображенні близькі по кольору. Крім того, ми можемо використовувати подібність між кольорними площинами R, G і B у наших алгоритмах, що дає можливість створити ще більш ефективні алгоритми. Таким чином, при створенні алгоритму компресії графіки ми використовуємо особливості структури зображення.

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація хмарної системи стискання та перетворення форматів зображень.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

- Огляд існуючих систем стискання та перетворення форматів зображень.
- Дослідження хмарної системи стискання та перетворення форматів зображень.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		4

– Програмна реалізація хмарної системи стискання та перетворення форматів зображень.

Об'єктом дослідження є процес стискання та перетворення форматів зображень.

Предметом дослідження є методи стискання та перетворення форматів зображень.

Методи дослідження базуються на методах комп'ютерної графіки, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод стискання та перетворення форматів зображень.
- Розроблено вітчизняний продукт стискання та перетворення форматів зображень, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі стискання та перетворення форматів зображень.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти «Наука – виробництву», 2023, основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №14.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація хмарної системи стискання та перетворення форматів зображень, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		5

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Розробляема, у результаті виконання магістерського проектування, система призначена для стиснення та перетворення форматів зображень. Формати графічних файлів визначають спосіб зберігання інформації у файлі (растровий або векторний), а також форму зберігання інформації (використовуваний алгоритм стиску).

Стиск застосовується для растрових графічних файлів, тому що вони мають звичайно досить великий обсяг. При стиску графічних файлів алгоритм стиску включається у формат графічного файлу.

Існують різні алгоритми стиску, причому для різних типів зображення доцільно застосовувати підходящі типи алгоритмів стиску.

Для стиску рисунків типу аплікації, що містять великі області однотонного зафарбування, найбільш ефективно застосування алгоритму стиску, що заміняє послідовність повторюваних величин (пікселів однакового кольору) на дві величини (піксель і кількість його повторень). Такий алгоритм стиску використовується в графічних файлах форматів BMP і PCX.

Для рисунків типу діаграми доцільне застосування іншого методу стиску, що використовує пошук повторюваних у рисунку «візерунків». Такий алгоритм використовується в графічних файлах форматів TIFF і GIF і дозволяє стиснути файл у кілька разів.

Для стиску відсканованих фотографій і ілюстрацій використовується алгоритм стиску JPEG. Цей алгоритм використовує той факт, що людське око дуже чутливе до зміни яскравості окремих крапок зображення, але набагато гірше зауважує зміну кольору. Дійсно, при глибині кольору 24 біта комп'ютер

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		6

забезпечує відтворення більше 16 млн. різних кольорів, тоді як людина навряд чи здатна розрізнити й тим більше назвати більше сотні кольорів і відтінків.

Застосування методу JPEG дозволяє стискати файли в десятки разів, однак може приводити до необоротної втрати інформації (файли не можуть бути відновлені в первісному виді).

Деякі формати графічних файлів є універсальними, тому що можуть бути оброблені більшістю графічних редакторів. Деякі програми обробки зображень використовують оригінальні формати, які розпізнаються тільки самою програмою, що створює. Перевага оригінальних форматів файлів полягає в тому, що вони дозволяють зберігати зображення при меншому розмірі файлу.

1.2 Область застосування

Областю застосування є робота з графічними файлами. Розглянемо деякі формати графічних файлів більш докладно.

Bit Map image (BMP) – універсальний формат растрових графічних файлів, використовується в операційній системі Windows. Цей формат підтримується багатьма графічними редакторами, у тому числі редактором Paint. Рекомендується для зберігання й обміну даними з іншими додатками.

Tagged Image File Format (TIFF) – формат растрових графічних файлів, підтримується всіма основними графічними редакторами й комп'ютерними платформами. Містить у собі алгоритм стиску без втрат інформації. Використовується для обміну документами між різними програмами. Рекомендується для використання при роботі з видавничими системами.

Graphics Interchange Format (GIF) – формат растрових графічних файлів, підтримується додатками для різних операційних систем. Включає алгоритм стиску без втрат інформації, що дозволяє зменшити обсяг файлу в кілька разів. Рекомендується для зберігання зображень, створюваних програмним шляхом (діаграм, графіків і так далі) і рисунків (типу аплікації) з обмеженою кількістю

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		7

кольорів (до 256). Використовується для розміщення графічних зображень на Web-сторінках в Інтернеті.

Portable Network Graphic (PNG) – формат растрових графічних файлів, аналогічний формату GIF. Рекомендується для розміщення графічних зображень на Web-сторінках в Інтернеті.

Joint Photographic Expert Group (JPEG) – формат растрових графічних файлів, що реалізує ефективний алгоритм стиску (метод JPEG) для відсканованих фотографій і ілюстрацій. Алгоритм стиску дозволяє зменшити обсяг файлу в десятки разів, однак приводить до необоротної втрати частини інформації. Підтримується додатками для різних операційних систем. Використовується для розміщення графічних зображень на Web-сторінках в Інтернеті.

Windows MetaFile (WMF) – універсальний формат векторних графічних файлів для Windows-додатків. Використовується для зберігання колекції графічних зображень Microsoft Clip Gallery.

Encapsulated PostScript (EPS) – формат векторних графічних файлів, підтримується програмами для різних операційних систем. Рекомендується для печатки й створення ілюстрацій у настільних видавничих системах.

CorelDraw files (CDR) – оригінальний формат векторних графічних файлів, використовуваний у системі обробки векторної графіки CorelDraw.

Якщо ви збираєтеся працювати із графічним файлом тільки в одному даному додатку, доцільно вибрати оригінальний формат. Якщо ж має бути передавати дані в інший додаток, інше середовище або іншого користувача, варто використовувати універсальний формат.

Таким чином, виходячи з вищеперерахованого, розробка програмного забезпечення хмарної системи стискання та перетворення форматів зображень, є актуальною задачею, яка потребує вирішення у даному магістерському проєкті.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		8

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень по профілю теми магістерського проекту

Розглянемо програми що дозволяють переглядати графічні файли й конвертувати їх в різні формати.

IrfanView

IrfanView – невеликий, але досить потужний додаток, призначений для перегляду й редагування графічних зображень. А додаткові модулі, які можна скачати із сайту розроблювача, значно розширюють список доступних функцій.

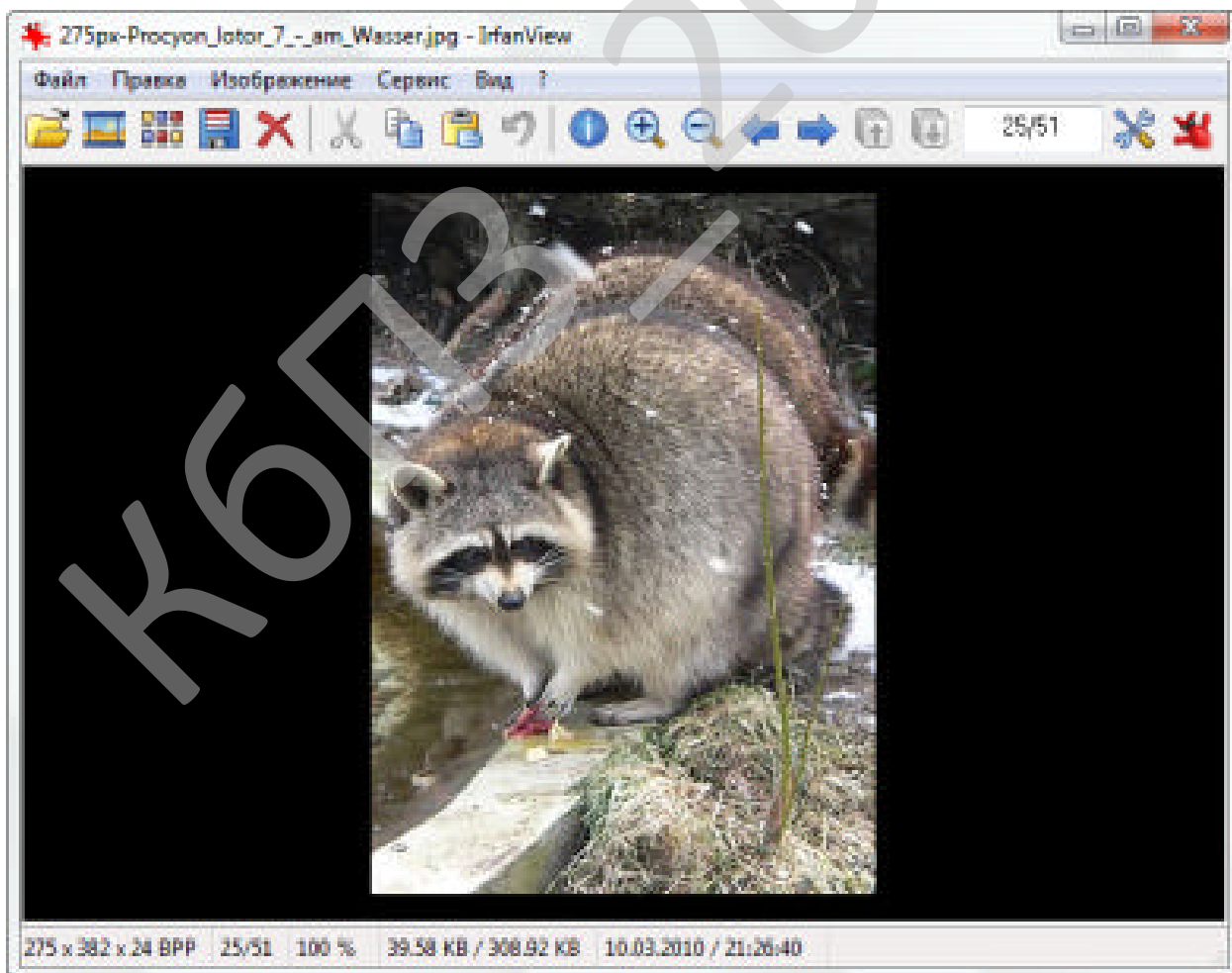


Рисунок 2.1 – Користувальницький інтерфейс IrfanView

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		9

По-перше, розглянемо можливості IrfanView як редактора зображень. Ці можливості не безмежні й значно уступають надпотужному Adobe Photoshop, однак безліч корисних операцій із графікою можна виконати безпосередньо в IrfanView, не переходячи в зовнішній редактор.

Відкривши файл у вікні IrfanView, користувач одержує можливість:

- Перетворити зображення в монохромне (одноколірне) або негативне.
- Зробити колірну корекцію: змінити параметри яскравості, контрастності, насиченості, колірного балансу й т.д.

- Поміняти місцями кольори у використовуваний колірній моделі. Наприклад, RGB (Red, Green, Blue) можна перетворити в BGR (Blue, Green, Red). У цьому випадку пікселі червоного кольору стануть синіми, а сині – червоними. У результаті вийде досить ефектна картинка.

- Змінити різкість зображення, а також застосувати більше десяти додаткових візуальних ефектів, у тому числі забрати "червоні очі" з фотографії, перетворити її в рисунок маслом і т.п.

- Змінити розміри або повернути зображення на необхідне число градусів.

- Виділити деяку область зображення, вставити в неї текст, видалити й виконати деякі інші операції.

Такі основні можливості IrfanView як графічного редактора. Друга сторона цієї програми – перегляд наявних зображень. Окрема функція відкриває вікно, у якому графічні файли зображені у вигляді мініатюр. З одного боку, нічого нового в цьому немає. Windows дозволяє бачити мініатюрні зображення й у своєму Провіднику, досить вибрати режим Ескізи сторінок. Але погано те, що Windows створює мініатюри не занадто високої якості. Це виправдано, оскільки прискорює вивід безлічі мініатюр на екран. Однак в окремих випадках хотілося б бачити ці картинки в максимально доступній якості. От для цих випадків і потрібний IrfanView.

Наступна часто, що вимагається функція, – перегляд наявних зображень у режимі слайд-шоу. Тут все просто: вибираємо папку, у якій зберігаються

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		10

зображення, указуємо, які типи файлів повинні брати участь у шоу, у якому порядку й з яким тимчасовим інтервалом їх потрібно виводити на екран, натискаємо кнопку Старт і любуємося результатом. Дуже зручно те, що будь-яке слайд-шоу можна зберегти у вигляді текстового файлу, щоб використовувати його надалі. Більше того, слайд-шоу може бути записане на компакт-диск у вигляді презентації або навіть збережена як екранна заставка. До речі, окрема функція програми дозволяє створювати із зображення шпалери для робочого стола.

IrfanView уміє виконувати пакетне перейменування й перетворення файлів з одного формату в іншій. При перейменуванні до кожного ім'я файлу може автоматично додаватися будь-який текст, а також число-лічильник. Пакетне перетворення з одного формату в іншій може супроводжуватися автоматичною зміною всіх параметрів зображення, на вибір користувача.

IrfanView підтримує роботу з TWAIN-пристроями, тому зображення може бути відскановано безпосередньо із цієї програми. Передбачено можливість створювати скріншоти (знімки екрана). При цьому користувач налаштовує наступні параметри:

- Робити знімок усього екрана, тільки активного вікна, або активного вікна без рядків заголовка й стану.
- Виконувати захват при натисканні гарячої клавіші (зазначеної користувачем) або періодично (через зазначений користувачем інтервал часу).
- Чи відображати курсор миші на знімку екрана.
- Показувати скріншот у вікні IrfanView або зберігати його на диску, у зазначеної користувачем папці, у тім або іншому форматі.

Крім усього перерахованого, IrfanView підтримує безліч додаткових модулів, що виконують всілякі завдання. От лише кілька прикладів таких плагінів (plugins):

- Effects містить набір додаткових візуальних ефектів для обробки зображень.

- Email дозволяє відправляти зображення по електронній пошті.
- IV-Player запускає програвач для прослуховування (перегляду) звукових і відео-файлів.

XnView Deluxe

Програма розроблена як комерційний продукт. Являє собою розширену версію безкоштовної програми XnView. У цей момент програма не розробляється й не підтримується, продаж і поширення програми також припинені. Програма не змогла зайняти значної ніші на ринку подібного програмного забезпечення. Проте, на даний момент багато функцій з Deluxe-версії перекочували в безкоштовну.

Відмінні риси XnView Deluxe від XnView:

- Керування проектами, альбомами й категоріями (планується перенести в XnView).
- Майстер експорту, корекції зображення й відправлення по email.
- Відображення HTML, PDF.
- Відображення SVG, Flash (потрібно встановлений плагін).
- Об'єднання зображень в PDF.
- Ефекти переходу й аудіо супровід у режимі показу слайдів.
- Експорт слайдів у відео, що виконується файл, CD, VCD або SVCD.
- Поліпшене відтворення MP3, Ogg Vorbis.
- Перегляд умісту архівів формату ZIP, RAR, CAB і ACE (Уже реалізовано у вигляді плагіна, розповсюджуваного автором безкоштовно – окремо або в складі XnView).
- Створення архівів.
- Відправлення файлу по email, FTP (функції FTP реалізовані у вигляді плагіна).
- Розширений пошук.
- Перегляд через контекстне меню.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		12

Pixillion 2.11

Pixillion 2.11 – нова версія безкоштовної програми, призначеної для конвертування графічних файлів. Pixillion підтримує формати GIF, jpg, pdf, png, bmp і інші. У цій версії додана підтримка нових форматів: pgf, raf, crw, cr2, kdc, dcr, mrw, nef, orf, dng, ptx, pef, arw, srf, x3f, erf, mef, mos, raw, r3d.

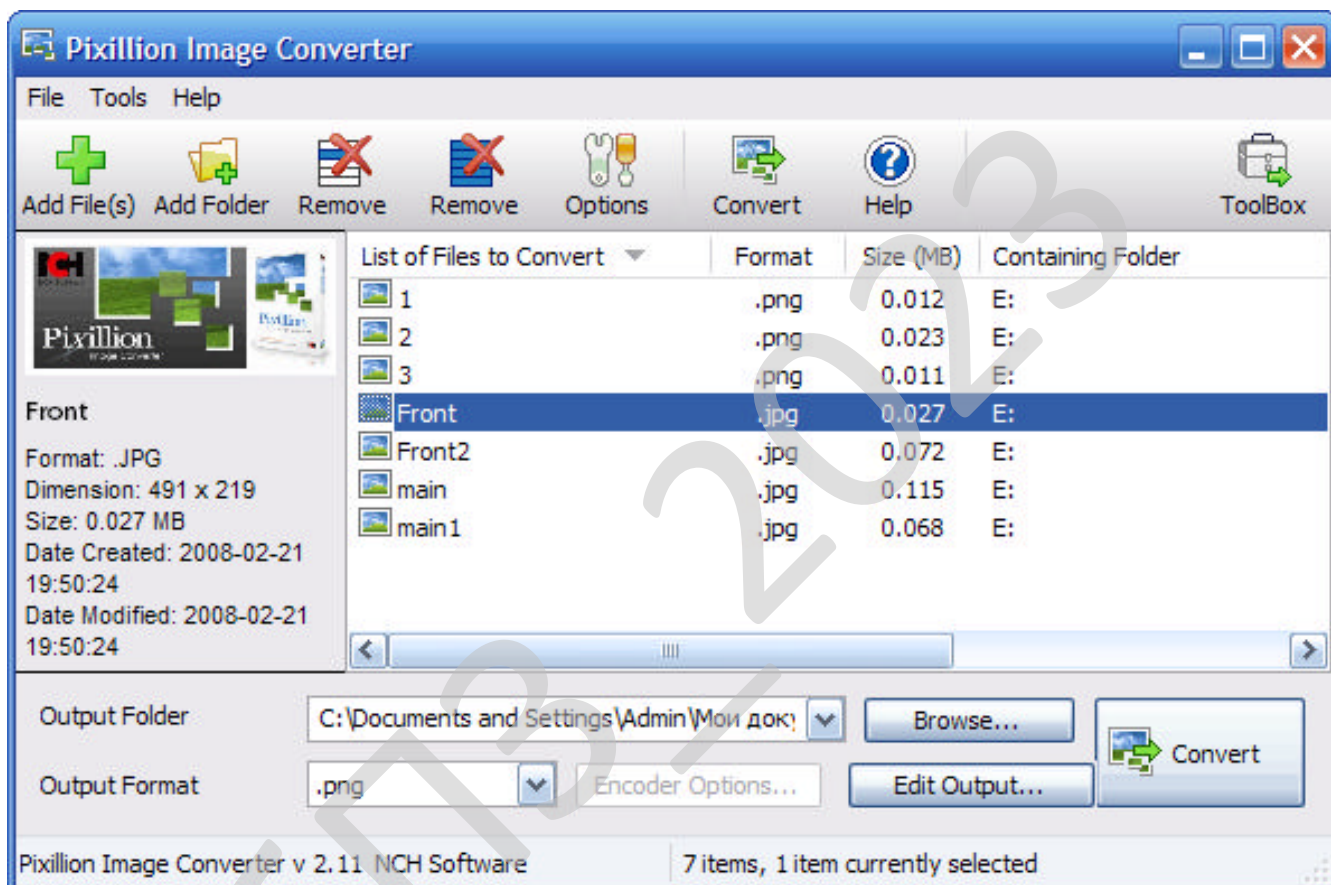


Рисунок 2.2 – Користувальницький інтерфейс Pixillion

Pixillion може завантажувати зображення безпосередньо з диска або із зазначеної папки. Перед виконанням перетворення кожен файл можна переглянути. У налаштуваннях Pixillion можна вказати параметри вихідного формату, а також визначити поведінку програми у випадку виникнення помилок або виявлення у вихідній папці файлів з ідентичними назвами. Для зручності Pixillion може бути інтегрована в контекстне меню Windows.

За твердженням розроблювачів Pixillion робить конвертацію графічних файлів набагато швидше, ніж аналогічні рішення від конкурентів. "Конвертація в один клік!" – такий слоган висить на сайті цієї програми.

VSO Image Resizer

Основне призначення програми складається в збільшенні й зменшенні розміру зображень. Однак, крім цієї функції тут присутня можливість зміни рівня стиску графічного файлу, конвертування в інший формат, установки водяного знака й пакетного перейменування імен файлів по встановленій масці.



Рисунок 2.3 – Користувальницький інтерфейс VSO Image Resizer

Унікальність цієї програми полягає в тому, що всі вищеписані дії можна зробити за один раз. VSO Image Resizer підтримує все графічні формати, що часто зустрічаються (JPEG, GIF, BMP, PNG і ін.), а також кілька форматів цифрових фотокамер (Canon – CR2, Nikon – NEF і MRW – Minolta). Програма інтегрується в Провідник Windows, забезпечуючи швидкий і зручний спосіб редагування параметрів зображення.

VSO Image Resizer – інструмент зроблений для тих, хто зберігає свої цифрові зображення й фотографії на ПК і хоче при цьому змінювати розміри, стискати, конвертувати, копіювати, імпортувати або впорядковувати фотографії.

З VSO Image Resizer Ви легко зможете створити копії зображень, придатні для пересилання по електронній пошті, завантажувати їх швидше, легко переміщати з папки в папку, змінювати їхній формат, редагувати великі номери файлів і в такий спосіб заощаджувати місце на Вашім жорсткому диску.

VSO Image Resizer інтегрується в оболонку провідника Windows, клацання правою кнопкою миші на Ваших зображеннях – і можна починати роботу.

Ключові особливості й характеристики VSO Image Resizer:

- Перетворення (конвертування) між різними графічними форматами.
- Коефіцієнт стиску, що налаштовується/розрішення.
- Підтримувані формати файлів: JPEG, GIF, BMP.
- Обробляє окреме зображення або працює в пакетному режимі.
- Зменшує розміри фотографій, призначених для публікацій в інтернеті або пересилання по електронній пошті.
- Інтегрується в провідник windows або працює як автономний додаток.
- Швидка обробка.
- Імпортує безпосередньо з Ваших карток пам'яті.
- Зберігає історію папки призначення.
- Змінювані варіанти й параметри настроювань для досвідчених користувачів.
- Підтримка мультимовного інтерфейсу.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		15

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Embarcadero Delphi, раніше Borland Delphi і Codegear Delphi, – інтегроване середовище розробки ПЗ для Microsoft Windows, Mac OS, iOS і Android мовою Delphi (що раніше носила назву Object Pascal), створена спочатку фірмою Borland і на даний момент приналежна й розроблювальна Embarcadero Technologies. Embarcadero Delphi є частиною пакета Embarcadero RAD Studio і поставляється в чотирьох редакціях: Community (поширюється безкоштовно й має обмежену ліцензію на використання в комерційних цілях), Professional, Enterprise і Architect.

Delphi 10.4 Sydney

Випущено 26 травня 2020 року. RAD Studio Delphi 10.4 забезпечує значно поліпшену високопродуктивну нативну підтримку Windows, кращу продуктивність розробки, миттєві підказки code completion, прискорення виконання коду із синтаксисом керованих записів, поліпшення виконання паралельних завдань на сучасних багатоядерних CPU, а також містить більш 1000 виправлень багів, поліпшення продуктивності середовища й бібліотек і багато чого крім того.

Основні можливості Delphi 10.4.1:

– Істотні розширення для Windows: поліпшення для застосунків на моніторах 4K High DPI, інтеграція з новим WebView2 на базі Chromium, використання розширених title bars, таких же, як в Office, Explorer, Google Chrome.

– Керування пам'яттю в Delphi тепер стандартизоване на всіх підтримуваних платформах – мобільних, настільних і серверних – використовувачи класичну реалізацію керування пам'яттю об'єктів.

– Істотне поліпшення Delphi Code Insight (без можливого блокування IDE – в окремому процесі), що допоможе при роботі з великими проектами.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		16

- Тип даних Delphi «record» тепер підтримуть довільні ініціалізацію, фіналізацію й операції копіювання.
 - Розширена підтримка бібліотек C++: ZeroMQ, SDL2, SOCI, libSIMDpp і Nematode.
 - Відладник Win 64 (на LLDB) і збирач для C++.
 - Поліпшення для C++: Включена велика кількість поліпшень STL з Dinkumware.
 - Підтримка Metal Driver GPU для macOS і iOS.
 - Вбудований Fmxlinux.
 - Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API. Реалізація компонента Media Player для macOS тепер використовує Avfoundation. Реалізований заново стилізуємий FMX компонент TМемо на платформі Windows значно поліпшений і тепер має відмінну підтримку ІМЕ.
 - Численні поліпшення швидкості й стабільності роботи нашої бібліотеки The Parallel Programming Library (PPL).
 - Додані оновлені драйвери для FireBird, PostgreSQL і SQLite.
 - Клієнтські бібліотеки HTTP і REST Client розширені застосунковими можливостями роботи з HTTPS. Також були розширені можливості підтримки Amazon AWS services
 - У технологію Visual LiveBindings внесена безліч поліпшень, у тому числі швидкодії, що стосуються, застосунків на VCL і FireMonkey
- RAD Studio 10.4 Короткий огляд:
- Істотні розширення для Windows. Створення застосунків, що чудово виглядають, із чіткими елементами інтерфейсу на 4к моніторах High DPI за допомогою нової гнучкої підтримки стилів елементів керування на екрані. Інтеграція із сучасними, безпечними web-технологіями від Microsoft – новим WebView2 на базі Chromium. Використання сучасних розширених title bars, таких же, як в Office, Explorer, Google Chrome, у своїх проектах. Істотні поліпшення надійності налагодження в новому відладнику для C++ Windows 64-bit.

						ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			17

Підвищення якості й швидкодії інструментів

- Велика кількість поліпшень STL від Dinkumware.
- Поліпшені деякі найважливіші методи й області RTL, на базі поліпшень сумісності з популярними бібліотеками C++.
- Поліпшена підтримка Cmake.
- Велика кількість виправлень для підвищення стабільності і якості.
- Відновлення Windows API – Обновлено й додали безліч декларацій API щоб добитися ще більшої інтеграції із платформою Windows.
- Загальні вдосконалення в бібліотеці доступу до БД FireDAC, включаючи оновлені драйвера для FireBird, PostgreSQL і SQLite. Вибір статичного або динамічного підключення SQLite до застосунку.

Змінені стилі VCL для High DPI

В 10.4, архітектура стилізації VCL була суттєво розширена для підтримки High DPI і 4K моніторів. Тепер усі елементи UI на формі VCL автоматично масштабуються під відповідне до монітора дозвіл для показу форми. Був оновлений API стилізації для підтримки стилів high DPI.

Кожний графічний елемент UI може бути обраний з наборів різних масштабів і масштабований до потрібного DPI, що дає чітке зображення елементів UI на всіх моніторах.

Нові High DPI стилі й стилізація окремих VCL компонент

Обновлено велике число вбудованих і преміальних VCL стилів для підтримки нового режиму стилізації High-dpi. Це дозволяє вам створювати застосунку з відмінним дизайном для всіх моніторів.

Розроблювачі VCL застосунків тепер можуть використовувати трохи VCL стилів на різних формах в одному застосунку або в різних компонентах на одній формі. Це також включає стилізацію компонентів загальною темою для платформи. Крім застосункової гнучкості використання стилів, це дозволяє використовувати нестилізуємі компоненти із зовнішніх бібліотек в VCL застосунках, що використовують стиль.

						ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			20

Поліпшена кроссплатформеність

- Додана підтримка Metal Driver GPU для macOS і iOS.
- Крім підтримки останнього iOS SDK, в RAD Studio 10.4 розроблювачі можуть задовольнити нові вимоги Apple до набору стартових екранів.
- Реалізований заново стилізуємий FMX компонент TМемо на платформі Windows значно поліпшений і тепер має відмінну підтримку ІМЕ.
- Користувачам редакцій Enterprise або Architect доступна повна інтеграція Fmxlinux з IDE для створення клієнтських застосунків Linux з GUI.
- Компонент Twebbrowser для iOS тепер реалізований на Wkwebview API.
- Реалізація компонента Media Player для macOS тепер використовує Avfoundation.

Оновлений менеджер пакетів Getit

Менеджер пакетів Getit в IDE був значно вдосконалений.

Дати випуску релізів пакетів тепер видні, і можливе сортування списку по цих датах; відбір тільки встановлених пакетів, контенту, доступного тільки при наявності підписки, багато чого іншого.

Універсальний інсталятор для установки Online і Offline

В 10.4 включений новий універсальний інсталятор, який використовує технологію на базі Getit. Цей інсталятор підтримує як online, так і offline (з ISO) варіанти установки.

Тепер обоє варіанта установки дозволяють вам указати початковий набір можливостей RAD Studio для установки, наприклад, свою комбінацію мов програмування й цільових платформ, мов інтерфейсу, і додавати до нього або видаляти непотрібне в будь-який момент.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випуск кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		21

забезпечення, яке призначено для хмарної системи стискання та перетворення форматів зображень.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі.

Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформулювати висновки про виконаний обсяг робіт та одержані результати.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		22

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Стиснення зображення – це процес, застосований до графічного файлу для мінімізації його розміру в байтах без погіршення якості зображення нижче прийняттого порогу. Зменшивши розмір файлу, можна зберігати більше зображень на заданому об'ємі диска чи пам'яті. Зображення також вимагає меншої пропускної здатності під час передачі через Інтернет або завантаження з веб-сторінки, що зменшує перевантаження мережі та пришвидшує доставку вмісту.

Які є два типи стиснення зображення?

Методи, які використовуються для стиснення файлів зображень, зазвичай належать до однієї з двох категорій: із втратами та без втрат. Стиснення із втратами зменшує розмір файлу зображення, остаточно видаляючи менш важливу інформацію, зокрема зайві дані. Стиснення з втратами може значно зменшити розмір файлу, але також може знизити якість зображення до рівня спотворення, особливо якщо зображення надто стиснене. Однак якість можна зберегти, якщо стиснення застосовано ретельно.

Однією з проблем стиснення з втратами є його незворотність. Після того, як його було застосовано до зображення, це зображення ніколи не можна відновити до початкового стану. Якщо стиснення з втратами багаторазово застосовується до того самого зображення, воно дедалі більше спотворюється. Тим не менш, стиснення з втратами даних виявилось цінною стратегією для Інтернету, де часто допускається помірне погіршення якості зображення.

Найпоширенішим прикладом стиснення з втратами є JPEG, формат стиснення зображень, який широко використовується в Інтернеті та цифровій

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		23

фотографії. Цей загальновизнаний формат підтримується численними інструментами та програмами. Крім того, стиснення можна застосовувати в градусах, що дає змогу використовувати стиснення JPEG, яке найкраще забезпечує баланс між розміром файлу та якістю.

Інший підхід до стиснення зображення називається без втрат. Цей метод застосовує стиснення без видалення критичних даних або зниження якості зображення, що призводить до стисненого зображення, яке можна відновити до початкового стану без погіршення чи спотворення. Однак стиснення без втрат не зменшує розмір файлу майже так само, як стиснення з втратами, пропонуючи невелику перевагу з точки зору місця для зберігання, пропускної здатності мережі або швидкості завантаження. Стиснення без втрат зазвичай використовується в ситуаціях, коли якість зображення важливіша за простір на диску чи продуктивність мережі, наприклад для зображень продуктів або для демонстрації творів мистецтва.

Одним із найпоширеніших форматів без втрат є PNG, широко використовуваний формат, який зменшує розмір файлу шляхом визначення шаблонів і стиснення цих шаблонів разом. Хоча файли PNG, як правило, більші за файли JPEG, веб-сайти широко використовують їх, коли потрібно більше деталей зображення, наприклад для логотипів, значків, скріншотів або зображень із текстом. Іншим знайомим форматом без втрат є BMP, власний підхід до стиснення зображень, запроваджений Microsoft і використовується переважно для продуктів Microsoft, зокрема для комп'ютерів Windows.

GIF – це формат стиснення, який відноситься до категорії без втрат, хоча існує певна плутанина щодо того, чи це формат стиснення, чи без втрат. Зображення GIF обмежені 256 кольорами, тому перетворення зображення з більшою кількістю кольорів у GIF призводить до втрати якості, що іноді пояснюється стисненням із втратами. Але алгоритми стиснення, які використовує GIF, є без втрат. Якщо якість втрачається, це пов'язано з проблемами, пов'язаними з перетворенням файлу. В даний час формат GIF

						ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата			24

використовується переважно для простих відео та анімацій.

Формат стиснення, який досягає успіху, – це WebP від Google, формат зображень, розроблений виключно для Інтернету. На відміну від більшості методів стиснення, WebP підтримує як стиснення без втрат, так і без втрат, що робить його надзвичайно універсальним. Зображення WebP зазвичай займають менше місця на диску, ніж інші формати, але мають порівнянну якість. Більшість основних браузерів підтримують зображення WebP.

Стиснення також можна використовувати для типів файлів, не пов'язаних із зображеннями, таких як текстові або програмні файли, але його використання зазвичай обмежується стисненням без втрат. У текстових і програмних файлах надзвичайно важливо, щоб стиснення було без втрат, оскільки одна помилка може пошкодити значення текстового файлу або спричинити невиконання програми. Формат файлу zip є прикладом стиснення без втрат, який зазвичай використовується для текстових файлів або навіть цілих каталогів файлів.

Під час стиснення зображення невелика втрата якості зазвичай непомітна. У ситуаціях, коли є певний допуск до втрати, до файлів можна застосувати більший рівень стиснення, ніж коли допуск до втрати відсутній. З цієї причини графічні зображення зазвичай можна стиснути набагато більше, ніж текстові або програмні файли.

Нижче ми розглядаємо всі основні формати графічних файлів, від растрових веб-зображень до векторів і файлів програмного забезпечення для редагування зображень.

Ми докладно вивчаємо переваги та недоліки, підтримку браузера й ОС, а також ідеальні випадки використання кожного формату.

- JPEG (та JPG) – Joint Photographic Experts Group.
- PNG – переносна мережева графіка.
- GIF – формат обміну графіками.
- WebP.
- TIFF.

- BMP – Bitmap.
- HEIF – високоефективний формат файлу зображень.
- SVG – масштабована векторна графіка.
- EPS – інкапсульований постскриптум.
- PDF – портативний формат документа.
- PSD – документ Photoshop.
- AI – Ілюстрація Adobe Illustrator.
- XCF – Експериментальний обчислювальний центр.
- INDD – документ Adobe InDesign.
- Типи файлів необроблених зображень

1. JPEG (і JPG) – Joint Photographic Experts Group

JPEG (або JPG) – це формат файлу растрового зображення зі стисненням із втратами, що робить його придатним для обміну зображеннями. JPEG є «з втратами», тобто вони зменшують розмір файлу, але також зменшують якість зображень, коли ви використовуєте цей формат.

JPEG досі залишається одним із найбільш використовуваних типів файлів зображень, які ви побачите в Інтернеті, завдяки його стисненню та практично універсальній підтримці браузера/ОС.

Більшість платформ соціальних мереж (наприклад, Facebook та Instagram) автоматично перетворюють завантажені файли зображень у формат JPEG. Вони також використовують унікальні розміри зображень у соціальних мережах, щоб контролювати роздільну здатність ваших фотографій.

Переваги та недоліки

- Підтримка універсального браузера та ОС.
- Досить малий розмір файлу.
- Стиснення зображень із втратами може призвести до поганої читабельності тексту.

Підтримка браузерів і ОС

- Підтримується з версії 1.0 усіма основними браузерами (Chrome,

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		26

Firefox, Safari тощо)

– Підтримується за замовчуванням усіма засобами перегляду зображень і редакторами всіх основних операційних систем.

Випадки використання

– Хороший вибір для зображень блогів і статей, як-от знімків голови співбесідників, зображень продуктів тощо.

– Не **використовуйте** JPEG для інфографіки з великою кількістю дрібного тексту або скріншотів підручників, де текст є ключовим.

2. PNG – портативна мережева графіка

PNG – це растровий графічний формат, який підтримує стиснення без втрат, зберігаючи деталізацію та контраст між кольорами.

Зокрема, PNG забезпечує набагато кращу читабельність тексту, ніж JPEG.

Це робить PNG більш популярним вибором для інфографіки, банерів, скріншотів та іншої графіки, яка містить як зображення, так і текст.

Переваги та недоліки

– Зображення вищої якості (без втрат) і чітко видимий текст.

– Великі розміри файлів можуть уповільнити роботу веб-сайту, якщо їх надмірно використовувати (особливо зображення з високою роздільною здатністю).

Підтримка браузерів і ОС

– Підтримується всіма основними браузерами (Chrome, Edge, Firefox, Internet Explorer, Opera, Safari).

– Підтримується всіма основними операційними системами та їхніми стандартними редакторами зображень.

Випадки використання

– Хороший вибір для інфографіки, банерів, графіки блогів, скріншотів, купонів та інших візуальних матеріалів, які містять текст.

– Не **використовуйте** для фотографій з високою роздільною здатністю, оскільки це призведе до створення великих файлів розміром до кількох мегабайт.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		27

3. GIF – формат обміну графікою

GIF – це ще один тип файлу зображення, який відноситься до растрового формату. Він використовує стиснення без втрат, але «обмежує» зображення 8 бітами на піксель і обмеженою палітрою з 256 кольорів.

Формат GIF є найвідомішим (і найчастіше використовуваним) для анімованих зображень, оскільки його 8-бітове обмеження зберігає розміри файлів анімації малими та зручними для Інтернету.

Переваги та недоліки

- Малий розмір файлу.
- Підтримка анімації.
- 8-бітове обмеження призводить до обмеження якості зображення.

Підтримка браузерів і ОС

- Підтримується всіма основними браузерами (Chrome, Edge, Firefox, Internet Explorer, Opera, Safari).
- Підтримується всіма основними операційними системами та їхніми стандартними редакторами зображень.

Випадки використання

- Використовуйте анімовані GIF-файли не лише для того, щоб «оживити вміст», але й продемонструвати, як виконувати завдання в посібниках і посібниках.
- Не використовуйте його, якщо вам потрібні більш ніж 8-бітні кольорові зображення (JPEG підтримує до 24-біт).

4. WebP

WebP – це формат зображень, розроблений спеціально для забезпечення кращого стиснення зображень без втрат і без втрат.

Перехід із JPEG і PNG на WebP може допомогти заощадити дисковий простір на сервері та значну пропускну здатність, завдяки зменшенню на 35% файлів зображень для однакової якості.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		28

Переваги та недоліки

- Менші файли для такої ж або кращої якості зображення.
- Підтримується не всіма браузерами та редакторами зображень.

Підтримка браузерів і ОС

- Google Chrome (версія 17+ для комп'ютера, 25+ для мобільних пристроїв), Firefox (65+), Edge (18+) і Opera (11.0+) підтримують WebP. Запланована версія Safari 14 підтримуватиме WebP.

- Формат досі не підтримується більшістю рідних редакторів зображень ОС, але професійні варіанти, такі як Photoshop, підтримують WebP.

Випадки використання

- Замініть файли JPEG і PNG, щоб зберегти пропускну здатність і прискорити роботу сайту. Якщо ви використовуєте WordPress як програмне забезпечення CMS, ось як використовувати зображення WebP у WordPress і доставляти інші формати до застарілих браузерів.

5. TIFF

Одним із форматів, який не можна пропустити в цьому посібнику щодо типів файлів зображень, є TIFF.

TIFF, що є скороченням від Tagged Image File Format, є форматом растрового зображення, який найчастіше використовується для зберігання та редагування зображень, які пізніше будуть використані для друку.

Хоча він підтримує стиснення з втратами, він зазвичай використовується як формат зображення без втрат. Крім того, більшість професійних графічних програм, які підтримують TIFF (Photoshop, Illustrator тощо), не використовують стиснення. Таким чином, зображення TIFF зазвичай мають великий розмір файлу.

Переваги та недоліки

- Високоякісні файли ідеально підходять для зберігання або друку .
- Великий розмір файлу через типове використання без стиснення.
- Обмежена підтримка браузера.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		29

Підтримка браузерів і ОС

– Жоден основний браузер не може відобразити файл TIFF без доповнень або розширень.

– В основному доступний як формат експорту для професійних інструментів для редагування та публікації зображень.

Випадки використання

– Зберігання та підготовка зображень і графіки до публікації.

– Використовується багатьма сканерами для збереження якості відсканованих документів або зображень.

6. BMP – Bitmap

Bitmap (BMP) – це переважно застарілий формат файлу зображення, який відображає окремі пікселі без стиснення або з невеликим його стисненням. Це означає, що файли BMP можуть легко стати надзвичайно великими, і їх непрактично зберігати або обробляти.

Переваги та недоліки

– Величезні розміри файлів без помітного покращення якості порівняно з такими форматами, як WebP, GIF або PNG.

Підтримка браузерів і ОС

– Підтримується всіма основними браузерами (Chrome, Edge, Firefox, Internet Explorer, Opera, Safari).

– Вбудована підтримка в більшості редакторів зображень ОС за замовчуванням, як-от MS Paint.

Випадки використання

– У 2023 році фактично немає законних випадків використання формату зображень BMP.

7. HEIF – високоефективний формат файлу зображення

HEIF, аббревіатура від High Efficiency Image File Format, – це формат зображення, розроблений командою, яка розробляє відеоформат MPEG, щоб стати прямим конкурентом JPEG.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		30

Теоретично стиснення майже вдвічі ефективніше, ніж JPEG, що забезпечує зображення подвійної якості з однаковими розмірами файлів.

Це растровий формат зображення, заснований на відображенні пікселів, тобто ви не можете масштабувати зображення без втрати якості.

Переваги та недоліки

- Відмінне співвідношення якості та розміру файлу.
- Відсутній браузер і обмежена підтримка ОС.

Підтримка браузерів і ОС

- Не підтримується жодним основним браузером.
- Вбудована підтримка в macOS Sierra та iOS 11 і новіших версіях, але не у відповідних версіях Safari.

Випадки використання

- Використовується деякими новішими телефонами та пристроями для зберігання фотографій у вищій якості, ніж файли JPEG.

8. SVG – масштабована векторна графіка

Формат файлу Scalable Vector Graphics, який зазвичай називають SVG, був розроблений W3C як мова розмітки для відтворення двовимірних зображень прямо в браузері.

Він не покладається на пікселі, як растровий формат, а використовує XML-текст для окреслення форм і ліній подібно до того, як математичні рівняння створюють графіки.

Це означає, що ви можете нескінченно масштабувати зображення SVG без втрати якості.

Переваги та недоліки

- Невеликі розміри файлів і масштабування без втрат для простих ілюстрацій, форм і тексту.
- Не ідеальний формат для зображень або складних малюнків.

Підтримка браузерів і ОС

- Підтримується всіма основними браузерами (Chrome, Edge, Firefox,

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		31

Internet Explorer, Opera, Safari).

– Редактори зображень за замовчуванням зазвичай не підтримують SVG (оскільки він не підходить для фотографій), але більшість програм для ілюстрації підтримують експорт SVG.

Випадки використання

– SVG – це ідеальний формат для логотипів, піктограм, простих ілюстрацій та всього іншого, що можна вільно масштабувати за допомогою адаптивного дизайну. Якщо ви використовуєте WordPress, обов'язково перевірте, як увімкнути підтримку WordPress SVG для вашого веб-сайту.

9. EPS – інкапсульований постскриптум

По суті, файл EPS (Encapsulated PostScript) – це файл векторного зображення, який використовується для зберігання ілюстрацій в Adobe Illustrator та іншому програмному забезпеченні для ілюстрації, наприклад CorelDraw.

Подібно до файлів SVG, EPS фактично є текстовим документом, який окреслює форми та лінії за допомогою коду, а не відображає пікселі та кольори. Як результат, файли EPS також підтримують масштабування без втрат.

Переваги та недоліки

- Масштабування без втрат.
- Підтримка принтера для документів і дизайнів.

Підтримка браузерів і ОС

– EPS не є стандартним форматом веб-файлів зображень і не підтримується жодним основним браузером.

Випадки використання

– Здебільшого використовується для зберігання, збереження та друку ілюстрацій під час роботи з Adobe Illustrator або іншим програмним забезпеченням.

10. PDF – портативний формат документів

Ймовірно, ви асоціюєте PDF зі збереженням, збереженням і читанням

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		32

переважно текстових документів. Це цілком зрозуміло – зрештою, документ є прямо там, у назві.

Але PDF-файли фактично базуються на тій самій мові PostScript, яка підтримує файли векторних зображень EPS, а також може використовуватися для збереження зображень та ілюстрацій.

Це обраний формат зображення для зберігання ілюстрацій, обкладинок журналів тощо для подальшого друку. Це також найкращий вибір для наших електронних книг Kinsta .

Переваги та недоліки

- Текст із можливістю індексування та пошуку робить його ідеальним для детальної інфографіки чи звітів.
- Може містити посилання, кнопки CTA та інші інтерактивні елементи.
- Масштабованість без втрат.
- Не можна включити до веб-вмісту, потрібно завантажити та прочитати як окремий файл .

Підтримка браузерів і ОС

- Підтримується всіма основними браузерами, але вам доведеться відкривати PDF як окремий файл. (Ви не можете використовувати файли PDF для відображення зображень у всьому вмісті HTML.)
- Підтримується як формат більшістю стандартних редакторів документів (наприклад, MS Word або Google Docs) і програмним забезпеченням для ілюстрації (AI, Inkscape), але не програмним забезпеченням для редагування зображень.

Випадки використання

- PDF – найкращий варіант, якщо ви хочете створити інтерактивний візуальний звіт або інфографіку, яка доповнює ваш вміст. Ось гарний список найкращих плагінів для перегляду PDF для користувачів WordPress.

11. PSD – документ Photoshop

Як впливає з назви, формат файлу PSD – це формат зображення, який

використовується для збереження документів із зображеннями та працює в Adobe Photoshop.

Це небезпечний для Інтернету формат зображень, тому він не підтримується жодними браузером чи стандартними засобами перегляду або редакторами зображень.

Переваги та недоліки

– Усі фільтри, прозорість, шляхи та редагування можна повністю налаштувати та відмінити.

– Якість зображення без втрат.

– Великі розміри файлів.

Випадки використання

– Збереження та зберігання проектів Photoshop до та після їх завершення.

12. AI – Ілюстрація Adobe Illustrator

AI – ще один формат зображення, спеціально розроблений Adobe для збереження не лише зображення, але й стану проекту.

Як і PSD, він не призначений для використання в Інтернеті та не підтримується ні браузерами, ні більшістю програм перегляду зображень за замовчуванням.

На відміну від файлів PSD, ви можете вільно масштабувати файли AI без втрати якості.

Переваги та недоліки

– Усі штрихи, лінії, фігури, фільтри можна налаштувати та оборотити.

– Збільшуйте або зменшуйте розмір ілюстрації вільно.

– Великі розміри файлів.

Випадки використання

– Збереження та зберігання проектів Adobe Illustrator до та після їх завершення.

13. XCF – Експериментальний обчислювальний центр

XCF, що розшифровується як eXperimental Computing Facility, – це тип

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		34

файлу зображення, який є рідним для редактора зображень з відкритим кодом GIMP . Це еквівалент файлів PSD і зберігає контури, прозорість, фільтри тощо.

Знову ж таки, це рідний тип файлу для зберігання проектів, і **він не підтримується жодним браузером** або програмою перегляду зображень за замовчуванням.

Переваги та недоліки

- Усі контури, фільтри та фігури можна налаштувати та відмінити.
- Якість зображення без втрат.
- Великі розміри файлів.

Випадки використання

- Збереження та зберігання проектів GIMP до та після їх завершення.

14. INDD – документ Adobe InDesign

INDD – це рідний тип файлу для користувачів Adobe InDesign, де можна зберігати файли проекту, зокрема вміст сторінки, стилі, зразки тощо.

Хоча його іноді називають форматом файлу зображення, він має тенденцію посилатися на візуальні елементи поза текстом.

Він не підтримується жодними веб-переглядачами чи програмами перегляду зображень за замовчуванням, оскільки **це небезпечний для Інтернету формат** .

Переваги та недоліки

- Зберігайте великі багатосторінкові дизайнерські проекти з настроюваними елементами.

Випадки використання

- В основному його використовують графічні дизайнери та художники-постановники для розробки та спільної роботи над плакатами, листівками, журналами, брошурами тощо.

15. Типи файлів необроблених зображень

Формати необроблених зображень – це типи файлів, які цифрова камера використовує для зберігання повноякісних зображень для подальшої обробки та

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		35

редагування.

Основні типи файлів необроблених зображень від виробника камери:

- Kodak: CR, K25, KDC
- Canon: CRW CR2 CR3
- Epson: ERF
- Nikon: NEF NRW
- Олімп: ORF
- Pentax: PEF
- Panasonic: RW2
- Sony: ARW, SRF, SR2

Замість 256 відтінків на колірний канал (8 біт), доступних у файлі JPEG, файли RAW пропонують до 16384 відтінків на колірний канал (14 біт) в одному зображенні. Це дає вам більше гнучкості під час налаштування кольорів і контрасту під час постобробки.

Необроблені зображення не призначені для Інтернету чи спільного використання та не підтримуються жодним основним браузером чи засобом перегляду зображень.

Переваги та недоліки

- Зображення вищої якості з більшою різноманітністю кольорів.
- Величезні файли зображень (необроблений файл може легко мати розмір від 20 до 40 мегабайт).

Випадки використання

- Збереження фотографій у найвищій якості для подальшої обробки та редагування.

На основі даних, пов'язаних із використанням Інтернету, три найпоширеніші типи файлів зображень: JPEG, PNG і SVG .

Нижче розглянемо, чому вони настільки поширені.

1. PNG: скріншоти, банери, інфографіка, зображення

Оскільки формат PNG краще підходить для тексту всередині зображення,

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		36

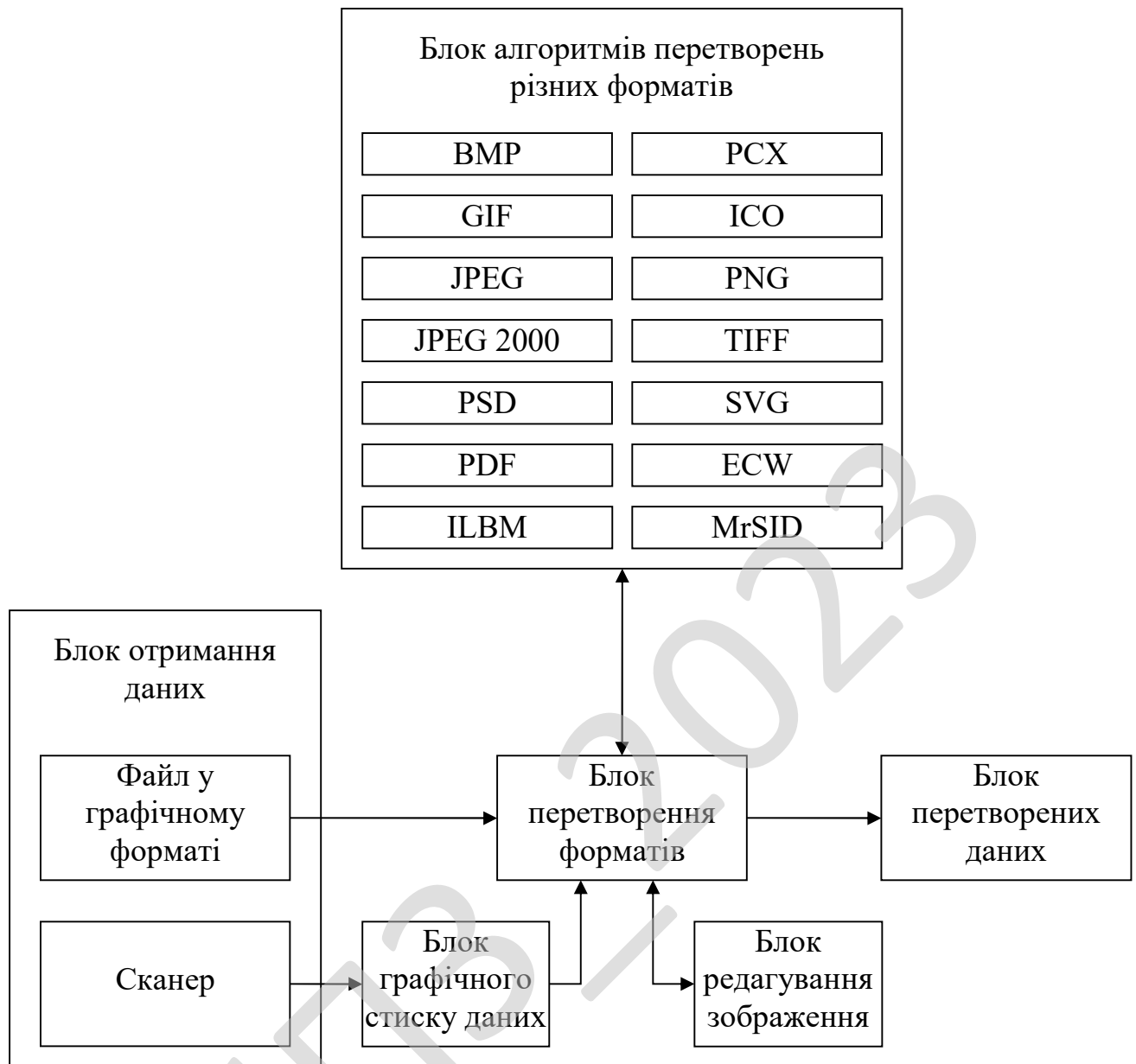


Рисунок 3.1 – Структурна схема системи

2. Блок алгоритмів перетворень різних форматів:

- Формат MrSID.
- Формат PSD.
- Формат SVG.
- Формат PDF.
- Формат BMP.
- Формат PCX.

- Формат GIF.
- Формат ICO.
- Формат JPEG.
- Формат JPEG 2000.
- Формат PNG.
- Формат TIFF.
- Формат ECW.
- Формат ILBM.

3. Блок перетворення форматів – призначений для перетворення графічних файлів з одного формату у інший.

4. Блок графічного стиску даних – призначений для стиснення даних, отриманих зі сканера, у графічний файл.

5. Блок редагування зображення – призначений для редагування зображення, тобто являє собою невеликий графічний редактор.

6. Блок перетворених даних – призначений для реалізації іншого алгоритму стиску даних, ніж той, що був, до перетворення.

3.3 Розробка функціональної схеми

На рисунку 3.2 зображена функціональна схема системи. Нижче розглянемо її більш докладно.

Функціональна схема складається з наступних блоків:

1. Головне вікно інтерфейсу користувача.
2. Панель меню користувача:
 - Файл.
 - Виправлення.
 - Зображення.
 - Ефекти.
 - Вид.



Рисунок 3.2 – Функціональна схема системи

- Масштаб.
- Утиліти.
- Вибране.

– Довідка.

3. Панель швидкого доступу до елементів програми:

– Відкрити зображення.

– Перетворити зображення.

– Зберегти зображення.

– Настроювання зображення.

– Збільшити зображення.

– Зменшити зображення.

– Роздрукувати зображення.

– Повернути зображення.

– Розтягти зображення.

– Інформація про розроблювача.

4. Блок функцій по конвертуванню графічних форматів з одного в іншій.

5. Блок функцій по редагуванню зображень:

– авто-обрізка;

– кадрування;

– видалення нахилу;

– фліп;

– дзеркальне відображення;

– зміна розміру зображення;

– поворот зображення;

– зрушення зображення;

– додавання шуму;

– вигин зображення;

– видалення границь.

6. Блок функцій по настроюванню зображень:

– зміна яскравості зображення;

– зміна колірного дозволу;

– подання зображення кольоровим або чорно-білим;

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		43

- зміна контуру;
- настроювання контрастності;
- настроювання розширення;
- видалити непотрібні пікселі;
- змінити кількість пікселів на дюйм.

7. Блок ефектів:

- тиснення;
- розмиття;
- гравірування;
- корекція гама;
- розмиття по гауссу;
- світіння;
- градієнт;
- відтінки сірого;
- півтону;
- контрастність;
- зміна кольору;
- застосування ефекту "імпресіонізм";
- інвертування зображення й тексту;
- видалення ліній;
- видалення сегментів ліній;
- максимальна фільтрація;
- медіани;
- мінімальна фільтрація;
- мозаїка;
- розмиття в русі;
- множення;
- ефект старих фотографій;
- пікселізація;

- полярність;
- постеризовування;
- панч, радіальний хвильовий;
- хвильовість;
- насиченість;
- різкість;
- гладкість.

8. Конвертація прямо з буфера обміну.

9. Зняття скріншотів.

10. Створення власного слайд-шоу з ефектами.

11. Блок пакетної обробки.

12. Підтримка декількох мов.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма взаємодії процесів системи, розробленої у результаті виконання магістерського проектування, наведена на рисунку 3.3.

Процеси взаємодіють наступним чином.

Спершу відбувається виведення головного вікна.

Цей процес взаємодіє з процесом відкриття зображень.

Процес відкриття зображень у свою чергу, взаємодіє з процесом виведення списку відкритих зображень. Процес виведення списку відкритих зображень взаємодіє з процесом вибору зображення зі списку. При виборі зображення зі списку можливий запуск наступних процесів:

- Збереження зображення.
- Виведення зображення.
- Зміна розмірів зображення.
- Стиснення зображення.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		45

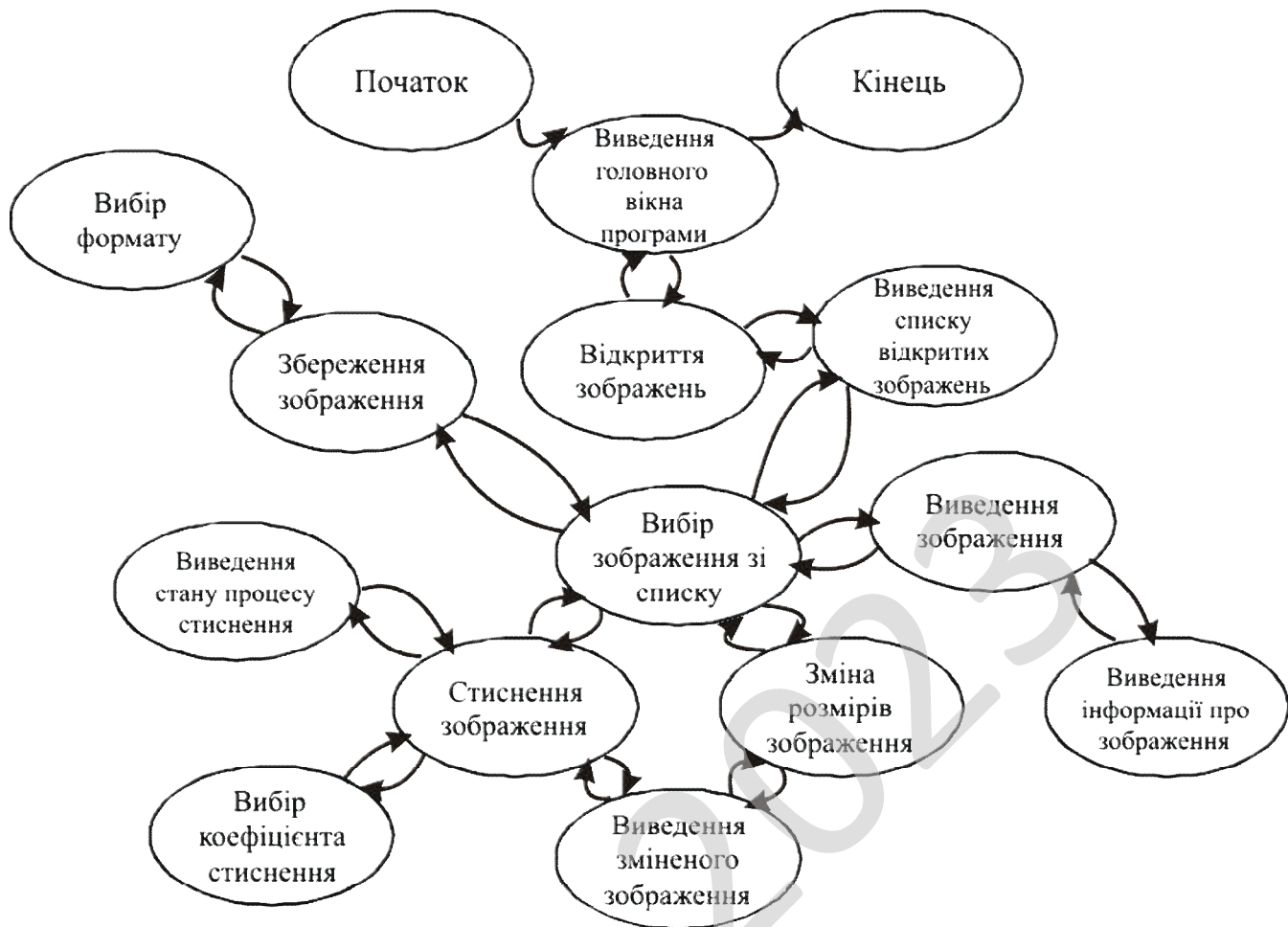


Рисунок 3.3 – Діаграма взаємодії процесів

Процес збереження зображення взаємодіє з процесом вибору формату, у якому потрібно зберегти зображення.

Процес виведення зображення взаємодіє з процесом виведення інформації про зображення.

Процес стиснення зображення взаємодіє з наступними процесами:

- Процес виведення стану процесу стиснення.
- Процес вибору коефіцієнту стиснення.
- Процес виведення зміненого зображення.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок–схеми та опис алгоритмів функціонування системи

На рисунку 4.1 наведено блок-схему основної програми. Її робота складається з виконання наступних кроків.

Спершу відбувається виведення основного вікна програми. Після цього користувач обирає відкрити зображення, або ні.

Якщо так, то відбуваються наступні дії:

- Відкривається файл з зображенням.
- Виводиться зображення.

У іншому випадку користувач стиснути зображення, або ні.

Якщо необхідно стиснути зображення, то виконуються наступні дії:

- Вибирається коефіцієнт стиснення.
- Запускається підпрограма стиснення зображення.
- Виводиться вікно стану процесу стиснення.
- Виводиться зображення, після стиснення.

Після цього користувач може обрати змінювати йому розмір зображення, або ні.

Якщо потрібно змінювати розмір зображення, то виконуються наступні дії:

- Обирається потрібний розмір зображення.
- Виводиться зображення після зміни розміру.

Користувач може обрати, чи потрібно йому зберегти змінене зображення.

Після цього користувач обирає працювати йому далі з системою, або ні.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		47

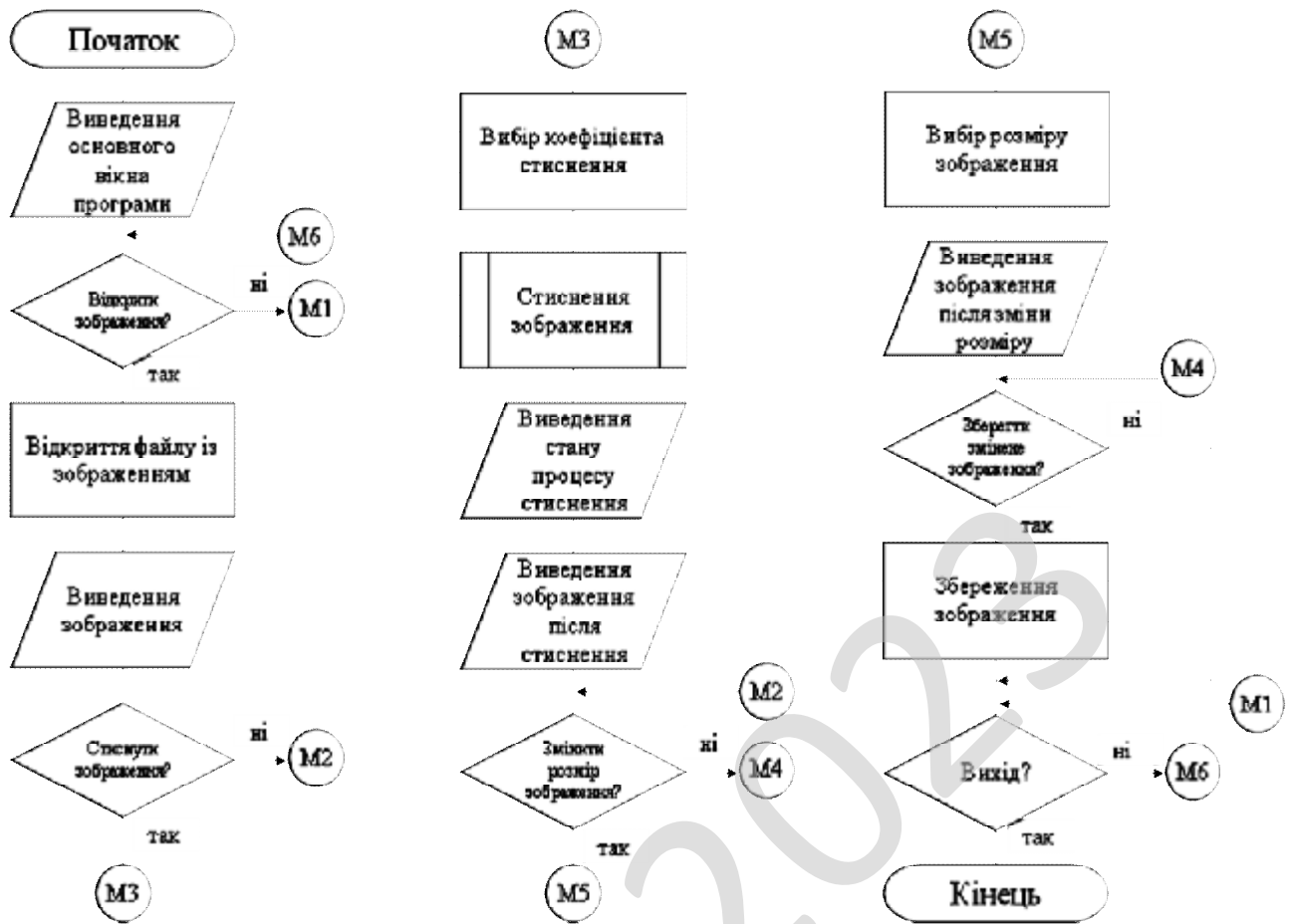


Рисунок 4.1 – Блок-схема роботи основної програми

На рисунку 4.2 зображена блок-схема підпрограми стиснення зображення.

Вона реалізована послідовним виконанням наступних етапів.

Спершу ініціалізується програмний процесор.

Відбувається читання файлу з зображенням.

Перетворюється RGB в YCrCb.

Застосовуються вейвлет перетворення, для кожної компоненти.

Відбувається ділення зображення на блоки 8x8.

Відбувається обробка усіх блоків наступним чином:

– Блок квантується.

– Відбувається кодування за Хаффманом.

Коли всі блоки оброблені, то відбувається повернення стиснутого зображення.


```

{$IFDEF WIN32}
type
  PPtrRec = ^TPtrRec;
  TPtrRec = record
    Lo: Word;
    Hi: Word;
end;
{$ENDIF}
function GetBigPointer(lp: pointer; Offset: Longint): Pointer;
begin
  {$IFDEF WIN32}
  GetBigPointer := @PByteArray(lp)^[Offset];
  {$ELSE}
  Offset := Offset + TPtrRec(lp).Lo;
  GetBigPointer := Ptr(TPtrRec(lp).Hi + TPtrRec(Offset).Hi *
  SelectorInc,
  TPtrRec(Offset).Lo);
  {$ENDIF}
end;
procedure TForm1.Button1Click(Sender: TObject);
var
  hPixelBuffer : THandle;
  lpPixelBuffer : pointer;
  lpPalBuffer : PLogPalette;
  lpBitmapInfo : PBitmapInfo;
  BitmapInfoSize : longint;
  BitmapSize : longint;
  PaletteSize : integer;
  i : longint;
  j : longint;
  OldPal : hPalette;
  hPal : hPalette;
  hBm : hBitmap;
  Bm : TBitmap;
  Dc : hdc;
  IsPaletteDevice : bool;
begin
  Application.ProcessMessages;
  {$IFOPT R+}
  {$DEFINE CKRANGE}
  {$ R-R-}
  {$ENDIF}

```

```

dc := GetDc(0);
IsPaletteDevice :=
GetDeviceCaps(dc, RASTERCAPS) and RC_PALETTE = RC_PALETTE;
{Give back the screen dc}
dc := ReleaseDc(0, dc);
{Розмір інформації про малюнок повинен рівнятися розміру BitmapInfo}
{плюс розмір таблиці квітів, мінус одна таблиця}
{тому що вона вже оголошена в TBitmapInfo}
BitmapInfoSize := sizeof(TBitmapInfo) + (sizeof(TRGBQUAD) * 255);
BitmapSize := (sizeof(byte) * 256) * 256;
{Розмір палітри повинен рівнятися розміру TLogPalette}
{плюс кількість осередків колірної палітри - 1, тому що}
{одна палітра вже оголошена в TLogPalette}
if IsPaletteDevice then
    PaletteSize := sizeof(TLogPalette) + (sizeof(TPaletteEntry) * 255);
    {Виділяємо пам'ять під BitmapInfo, PixelBuffer, i Palette}
    GetMem(lpBitmapInfo, BitmapInfoSize);
    hPixelBuffer := GlobalAlloc(GHND, BitmapSize);
    lpPixelBuffer := GlobalLock(hPixelBuffer);
if IsPaletteDevice then
    GetMem(lpPalBuffer, PaletteSize);
    {Заповнюємо нулями BitmapInfo, PixelBuffer, i Palette}
    FillChar(lpBitmapInfo^, BitmapInfoSize, #0);
    FillChar(lpPixelBuffer^, BitmapSize, #0);
if IsPaletteDevice then
    FillChar(lpPalBuffer^, PaletteSize, #0);
    {Заповнюємо структуру BitmapInfo}
    lpBitmapInfo^.bmiHeader.biSize := sizeof(TBitmapInfoHeader);
    lpBitmapInfo^.bmiHeader.biWidth := 256;
    lpBitmapInfo^.bmiHeader.biHeight := 256;
    lpBitmapInfo^.bmiHeader.biPlanes := 1;
    lpBitmapInfo^.bmiHeader.biBitCount := 8;
    lpBitmapInfo^.bmiHeader.biCompression := BI_RGB;
    lpBitmapInfo^.bmiHeader.biSizeImage := BitmapSize;
    lpBitmapInfo^.bmiHeader.biXPelsPerMeter := 0;
    lpBitmapInfo^.bmiHeader.biYPelsPerMeter := 0;
    lpBitmapInfo^.bmiHeader.biClrUsed := 256;
    lpBitmapInfo^.bmiHeader.biClrImportant := 256;
    {Заповнюємо таблицю квітів BitmapInfo відтінками сірого: від чорного до
білого}
for i := 0 to 255 do
begin

```

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		51


```

i, a: integer;
bm: TBitmap;
begin
  a := 0;
  bm := TBitmap.Create;
  image1.Picture.LoadFromFile(listbox1.Items[0]);
  bm.Height := image1.Height;
  bm.Width := listbox1.Items.Count * image1.Picture.Width;
  for i := 0 to listbox1.Items.Count - 1 do
    begin
      image1.Picture.LoadFromFile(listbox1.Items[i]);
      bm.Canvas.Draw(a, 0, image1.Picture.Graphic);
      a := a + image1.Picture.Height;
    end;
  //form1.Canvas.Draw(0,0,bm);
  bm.SaveToFile(Edit1.Text + '.bmp');
  bm.free;
end;

```

З файлів GIF (анімованих) витягаємо кожну картинку окремо, або записуємо в окремий BMP по черзі.

```

procedure TForm1.Button4Click(Sender: TObject);
var
  i, a: integer;
  bm: TBitmap;
begin
  a := 0;
  bm := TBitmap.Create;
  bm.Height := RxGIFAnimator1.Height;
  bm.Width := RxGIFAnimator1.Image.Count * RxGIFAnimator1.Width;
  for i := 0 to RxGIFAnimator1.Image.Count - 1 do
    begin
      RxGIFAnimator1.FrameIndex := i;
      bm.Canvas.Draw(a, 0, RxGIFAnimator1.Image.Frames[i].Bitmap);
      a := a + RxGIFAnimator1.Height;
    end;
  //form1.Canvas.Draw(0,0,bm);
  bm.SaveToFile(Edit1.Text + '.bmp');
  bm.free;
end;

```

Робота з JPEG зображенням

У комплект поставки Delphi входить модуль JPEG. Він дозволяє працювати із зображеннями у форматі JPEG. Ця програма відкриває обраний файл і виводить зображення на форму.

```
uses JPEG;  
  
procedure TForm1.Button1Click(Sender: TObject);  
var  
    JpegIm: TJpegImage;  
    bm: TBitmap;  
  
begin  
    if OpenDialog1.Execute = false then  
        Exit;  
    bm := TBitmap.Create;  
    JpegIm := TJpegImage.Create;  
    JpegIm.LoadFromFile(OpenDialog1.FileName);  
    bm.Assign(JpegIm);  
    Form1.Canvas.Draw(0, 0, bm);  
    bm.Destroy;  
    JpegIm.Destroy;  
end;
```

Для вставляння зображення у форматі JPEG у модуль, що виконується, в Delphi передбачено два класи TJpegImage і TJpegData. Ми будемо використовувати перший, він описаний у модулі JPEG (його треба підключити в uses).

Тепер зробимо конвертер картинок у форматі *.bmp у формат *.JPEG. Для цього нам знадобляться такі компоненти: TImage (для перегляду картинок), дві кнопки TButton (для відкриття діалогу вибору картинок і для запуску процесу), TTrackBar (для того щоб установлювати якість картинки), TCheckBox (щоб установлювати або забирати прапор "Відтінки сірого") і TOpenDialog.

Оброблювач події OnClick для першої кнопки може мати такий вид:

```
procedure TForm1.Button1Click(Sender: TObject);  
begin  
    if opendialog1.execute then  
        image1.Picture.LoadFromFile(opendialog1.filename);  
end;
```

									Арк.
									54
Вим.	Арк.	№ докум.	Підпис	Дата	ВКРМ-122.23.0027.00.00.ПЗ				

Необхідно настроїти фільтр для OpenDialog1, щоб можна було відкривати тільки картинки у форматі *.bmp.

Тепер безпосередньо займемося написанням основної частини програми, тобто створення JPEG-зображення. Всі дії будемо робити по клацанню на другій кнопці. Нам необхідно буде створити об'єкт типу TjpegImage, провести з ним необхідні дії, а потім за допомогою методу Compress, упакувати зображення й залишається тільки зберегти зображення у файл. Ще необхідно настроїти властивості TrackBar'a: властивість Max треба зробити рівною 100 і властивість Position рівною також 100. Отже, оброблювач натискання на другу кнопку стає таким:

```
procedure TForm1.Button2Click(Sender: TObject);  
var  
    jpg: TjpegImage;  
begin  
    {створюємо екземпляр об'єкта}  
    jpg := TjpegImage.Create;  
    {привласнюємо йому зображення}  
    jpg.Assign(image1.picture.graphic);  
    {установлюємо ступінь стиску (якість) 1..100}  
    jpg.CompressionQuality := TrackBar1.Position;  
    {якщо встановлено прапор "Відтінки сірого", то нехай картинка буде сірою}  
    jpg.Grayscale := checkbox1.Checked;  
    {Упаковуємо графіку}  
    jpg.Compress;  
    {і зберігаємо її куди захочеться}  
    jpg.SaveToFile('D:\first.jpg');  
    {знищуємо екземпляр об'єкта}  
    jpg.free;  
end;
```

Приведу повний код додатка:

```
unit Unit1;  
interface  
uses  
    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
    ComCtrls, ExtCtrls, StdCtrls, JPEG;  
type  
    TForm1 = class (TForm)  
        Button1: TButton;
```

```

    Button2: TButton;
    Image1: TImage;
    TrackBar1: TTrackBar;
    OpenDialog1: TOpenDialog;
    CheckBox1: TCheckBox;
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
private
    { Private declarations }
public
    { Public declarations }
end;
var
    Form1: TForm1;
implementation
    {$R *.DFM}
procedure TForm1.Button1Click(Sender: TObject);
begin
    if opendialog1.execute then
        image1.Picture.LoadFromFile(opendialog1.filename);
end;
procedure TForm1.Button2Click(Sender: TObject);
var
    jpg: TJpegImage;
begin
    {створюємо екземпляр об'єкта}
    jpg := TJpegImage.Create;
    {привласнюємо йому зображення}
    jpg.Assign(image1.picture.graphic);
    {установлюємо ступінь стиску (якість) 1..100}
    jpg.CompressionQuality := TrackBar1.Position;
    {якщо встановлено прапор "Відтінки сірого", то нехай картинка буде сірою}
    jpg.Grayscale := checkbox1.Checked;
    {Упаковуємо графіку}
    jpg.Compress;
    {і зберігаємо її куди захочеться}
    jpg.SaveToFile('D:\first.jpg');
    {знищуємо екземпляр об'єкта}
    jpg.free;
end;
end.

```

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		56

Перетворення JPEG в BMP

```
uses
    JPEG;

procedure JPEGtoBMP(const FileName: TFileName);
var
    JPEG: TJPEGImage;
    bmp: TBitmap;
begin
    JPEG := TJPEGImage.Create;
    try
        JPEG.CompressionQuality := 100; {Default Value}
        JPEG.LoadFromFile(FileName);
        bmp := TBitmap.Create;
        try
            bmp.Assign(JPEG);
            bmp.SaveToFile(ChangeFileExt(FileName, '.bmp'));
        finally
            bmp.Free
        end;
    finally
        JPEG.Free
    end;
end;
```

Збереження зображення у форматі JPEG

Ця функція зберігає зображення екрана у файлі C:\Screen.jpg.

```
uses JPEG;

procedure TForm1.Button1Click(Sender: TObject);
var
    JpegIm: TJpegImage;
    bm: TBitMap;
begin
    bm := TBitMap.Create;
    bm.Width := Screen.Width;
    bm.Height := Screen.Height;
    BitBlt(bm.Canvas.Handle, 0, 0,
    bm.Width, bm.Height,
    GetDC(0), 0, 0, SRCCOPY);
    JpegIm := TJpegImage.Create;
    JpegIm.Assign(bm);
    JpegIm.CompressionQuality := 20;
    JpegIm.Compress;
    JpegIm.SaveToFile('C:\Screen.jpg');
```

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		57

```
bm.Destroy;  
JpegIm.Destroy;  
end;
```

4.2 Захист розробленого програмного забезпечення

Дані які використовуються у даній роботі захищаються алгоритмом ДСТУ 7564:2014 («Купина»). В Україні на основі консервативного підходу із залученням відомих і добре досліджених конструкцій була розроблена геш-функція, що базується на новому блоковому шифрі „Калина” (ДСТУ 7624:2014).

Національний стандарт ДСТУ 7564:2014 визначає криптографічну функцію гешування „Купина”, додатковий режим її застосування для формування коду автентифікації повідомлення (імітовставки), а також значення для перевірки реалізацій.

Для скорочення обсягу тексту національного стандарту була застосована математична нотація, що дозволяє отримати точний і компактний запис.

Водночас, такий підхід може ускладнювати сприйняття сутності алгоритму для фахівців, що не мають фундаментальної криптологічної освіти.

У роботі наводиться розгорнутий альтернативний опис функції гешування „Купина” із позначеннями, традиційними для галузі комп’ютерних наук.

Термінологія та позначення

Вектор ініціалізації – бітова послідовність фіксованої довжини (512 або 1024 біта), що використовується як початкове значення при обчисленні геш-значення.

Внутрішній стан – бітова послідовність фіксованої довжини (512 або 1024 біта), що є проміжним значенням на кожній ітерації перетворення функції гешування, а також вхідним та вихідним значеннями перетворень P і Q ; для цих перетворень внутрішній стан подається як матриця розміром $8 \times c$ байт.

Геш-значення (геш-вектор) – бітова послідовність фіксованої довжини ($n = 8 \cdot s, s \in \{1, 2, \dots, 64\}$), що є результатом роботи функції гешування.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		58

Доповнення – вставка додаткових біт у кінець повідомлення для отримання кратності довжини бітової послідовності довжині внутрішнього стану функції гешування.

Повідомлення – бітова послідовність довжини від 0 біт (порожній рядок) до $2^{96}-1$ біт.

Функція стиснення – ітеративне перетворення, що відображає l -бітний блок повідомлення та l -бітне значення, отримане функцією стиснення на попередньому кроці, у нове l -бітне значення.

Далі використовуються наступні позначення:

- \oplus – додавання за модулем 2 (XOR);
- $0x$ – префікс числа, що записане у шістнадцятковій системі числення;
- $a \bmod b$ – ціле невід’ємне число, що дорівнює залишку від ділення цілого числа a на натуральне число b ;
- B_i – i -й байт вхідної послідовності;
- C^i – константа перетворення XORRoundKey або Add64RoundKey для i -го циклу;
- c – кількість стовпців внутрішнього стану в матричному поданні;
- ϕ – функція стиснення;
- H – визначена у стандарті функція гешування;
- $H(M)$ – результат обчислення функції гешування для повідомлення M (гешзначення);
- IV – вектор ініціалізації;
- l – розмір внутрішнього стану функції гешування (у бітах), $l \in \{512, 1024\}$;
- M – повідомлення;
- m_i – i -й блок повідомлення M ;
- n – довжина обчисленого геш-значення;
- N – довжина повідомлення M без доповнення;
- P, Q – складові перетворення функції стиснення;

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		59

- P_{512} – перетворення P для 512-бітного внутрішнього стану;
- P_{1024} – перетворення P для 1024-бітного внутрішнього стану;
- Q_{512} – перетворення Q для 512-бітного внутрішнього стану;
- Q_{1024} – перетворення Q для 1024-бітного внутрішнього стану;
- r – кількість ітерацій у перетвореннях P і Q ($r = t$ в ДСТУ 7564:2014);
- S – внутрішній стан геш-функції;
- t – кількість блоків m , з яких складається повідомлення M , включаючи доповнення;
- v_i – i -й біт вхідної послідовності;
- Ω – завершальне перетворення;
- Купина- n – режим використання функції гешування з усіченням обчисленого гешзначення до розміру n біт.

Загальні положення

Під функцією гешування H розуміється залежно від вектора ініціалізації IV відображення послідовності біт M у геш-значення $H(M)$ фіксованої довжини n .

ДСТУ 7564:2014 визначає функцію гешування, яка виконує перетворення «Купина-256» або «Купина-512», що забезпечують обчислення геш-значення з довжинами 256 або 512 біт відповідно.

Геш-значення довжиною 256 бітів додатково може бути усічено до бітової послідовності довжиною від 8 до 248 біт з кроком у 8 біт, 512 бітів може бути усічене до бітової послідовності довжиною від 264 до 504 біт з кроком у 8 біт.

Режим роботи для формування геш-значення довжиною n біт позначається як «Купина- n ». Основними режимами роботи функції гешування, що рекомендуються до застосування, є «Купина-256», «Купина-384» і «Купина-512».

Структура перетворення

Функція гешування, визначена в ДСТУ 7564:2014, формує геш-значення для повідомлення, що складається з бітової послідовності довжини від 0 біт (порожній рядок) до $2^{96}-1$ біт.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		60

При формуванні геш-значення повідомлення доповнюється, далі поділяється на l -бітні блоки m_0, \dots, m_t , після чого виконується обробка кожного блоку шляхом ітеративного виконання функції стиснення φ .

При цьому формуються значення $h_i = \varphi(h_{i-1}, m_i)$ де $i = 1, \dots, t$, а початкове значення $h_0 = IV$.

Після обробки останнього блоку повідомлення результуюче геш-значення обчислюється як $H(M) = \Omega(h_t)$, де Ω – завершальне перетворення, що повертає n - бітне значення, кратне 8 ($0 < n \leq l/2$).

КБГПЗ - 2023

					VKPM-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		61

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

На рисунку 5.1 приведено головне вікно програми.

З нього ми бачимо, на головній формі є наступні закладки:

- Файл.
- Виправлення.
- Зображення.
- Ефекти.
- Вид.
- Масштаб.
- Утіліти.
- Вибране.
- Довідка.

Крім цього є поле список зображень, яке дозволяє обрати зображення, та виконати наступні дії:

- Додати файл.
- Видалити файл.
- Переглянути файл.

Наступним є поле параметрів стиснення, яке дозволяє встановити якість зображення, та визначити розмір зображення.

Розмір може бути або початковим, або одним з заданих розмірів.

Крім того на формі наведені параметри збереження, які дозволяють виконати наступні дії над файлами:

- Замінити файл.
- Копіювати в папку.
- Упакувати в ZIP-файл.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		62

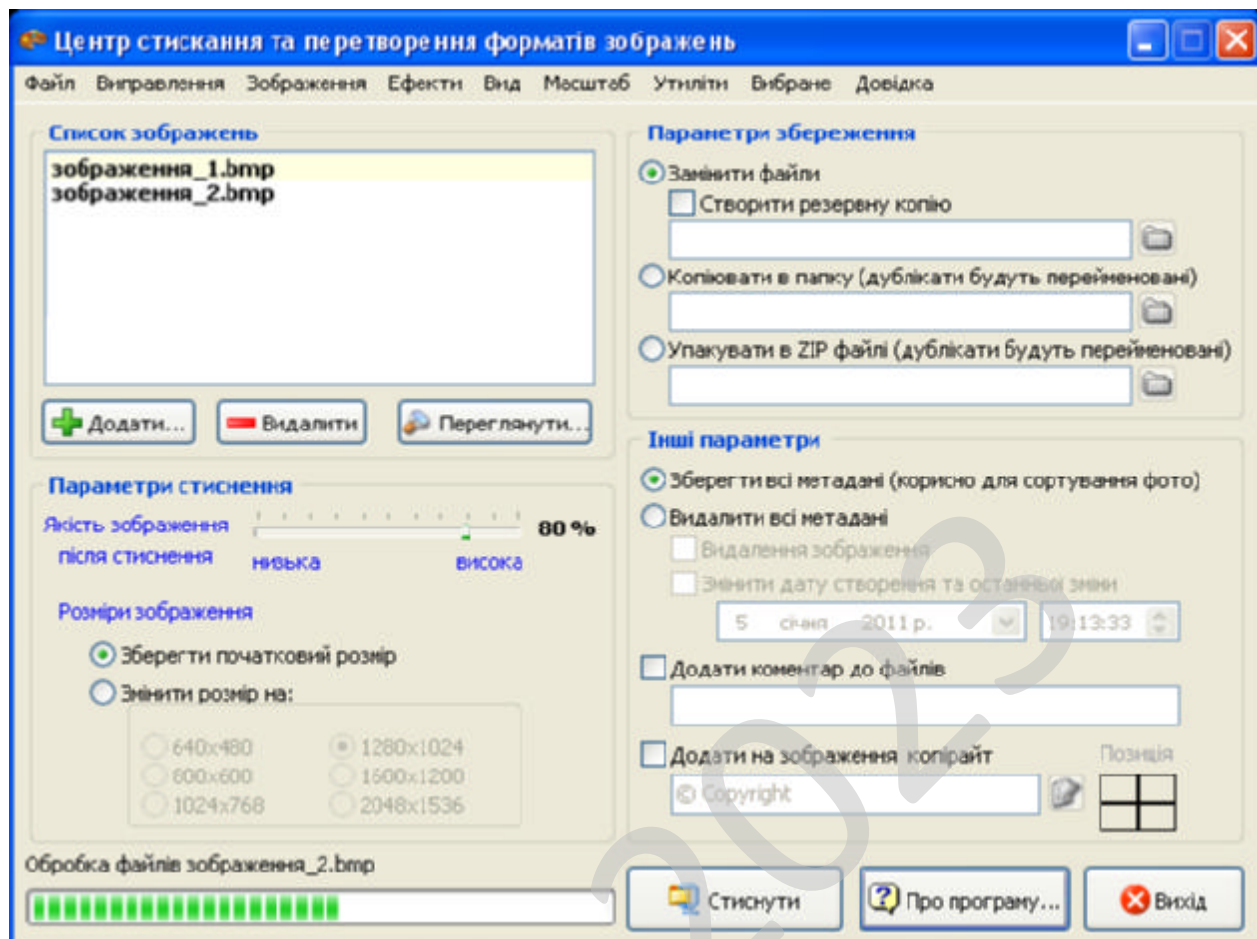


Рисунок 5.1 – Головне вікно програми

Крім того існують й інші параметри, які користувач може задати при роботі з зображеннями.

До них відносяться наступні параметри:

- Зберегти усі метадані.
- Видалити усі метадані.
- Додати коментар до файлів.
- Додати на зображення копірайт.

Також, на головну форму винесені кнопки швидкого доступу до наступних операцій:

- Стиснути.
- Про програму.
- Вихід.

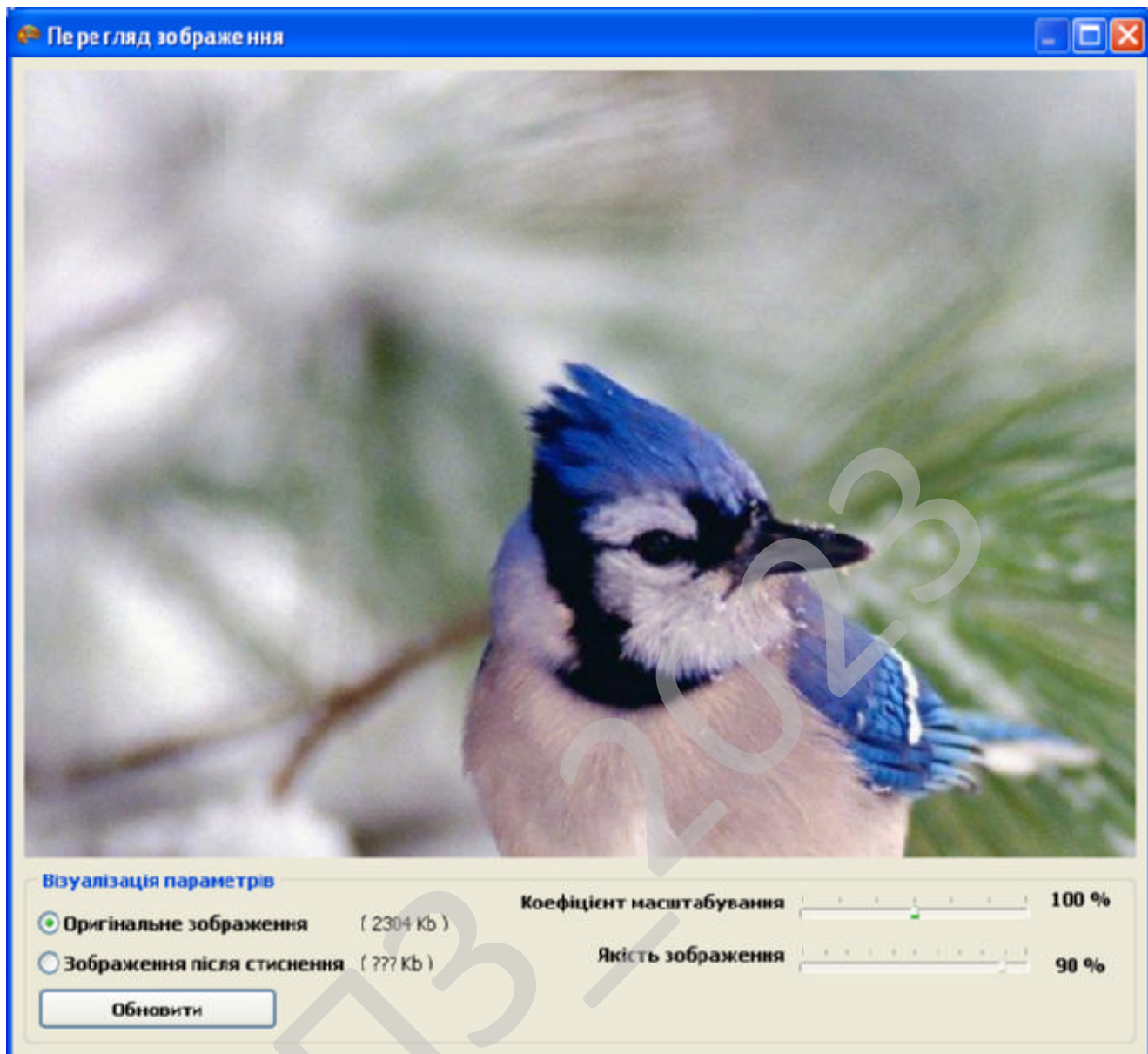


Рисунок 5.2 – Початкове зображення

На рисунку 5.2 наведено початкове зображення, з яким будуть проводитися наступні операції:

- Стиснення зображення на 25% (рисунок 5.3).
- Стиснення зображення на 90% (рисунок 5.4).

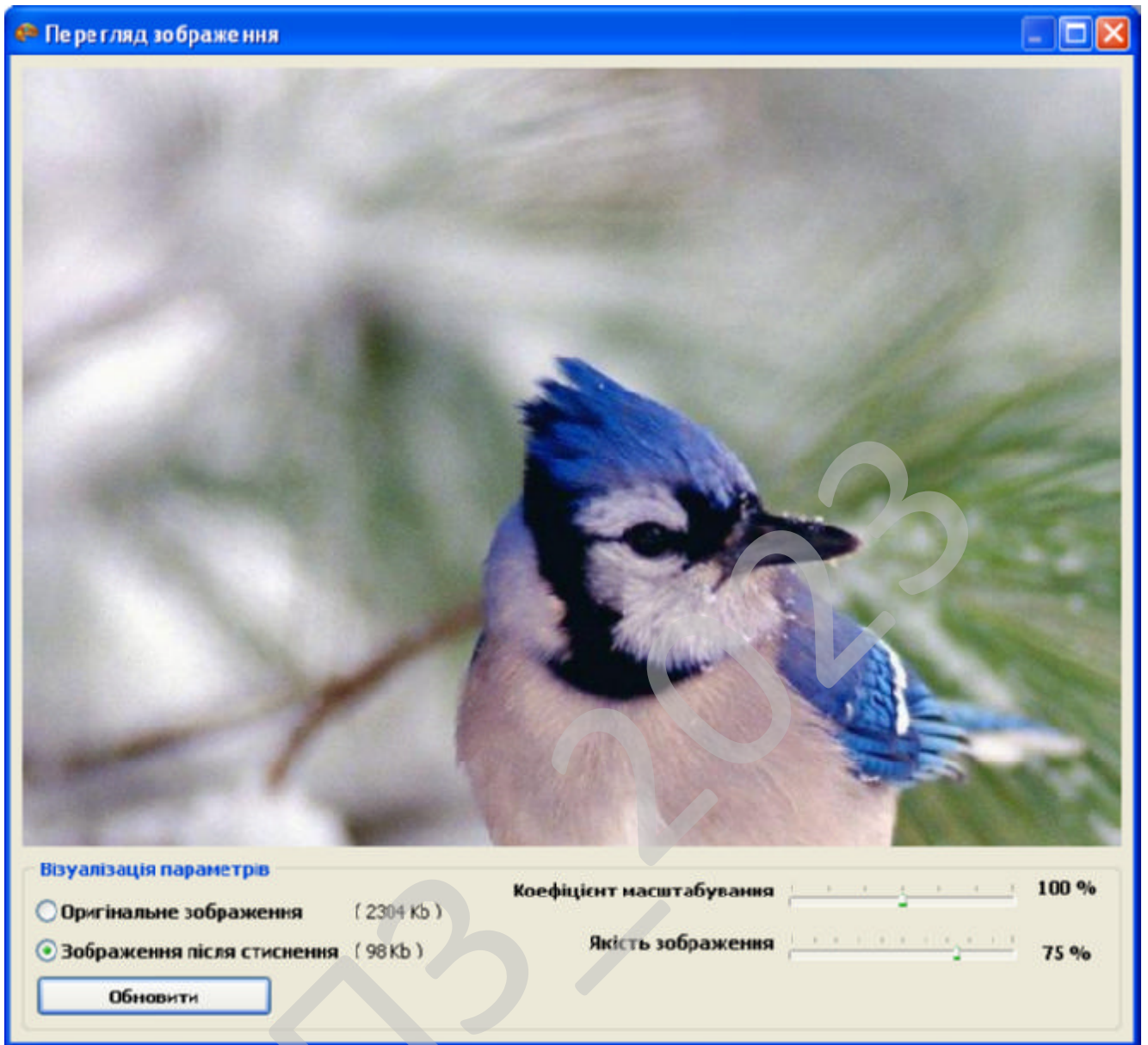


Рисунок 5.3 – Зображення після стиснення, якість зображення 75% (зменшення якості непомітно)

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		65

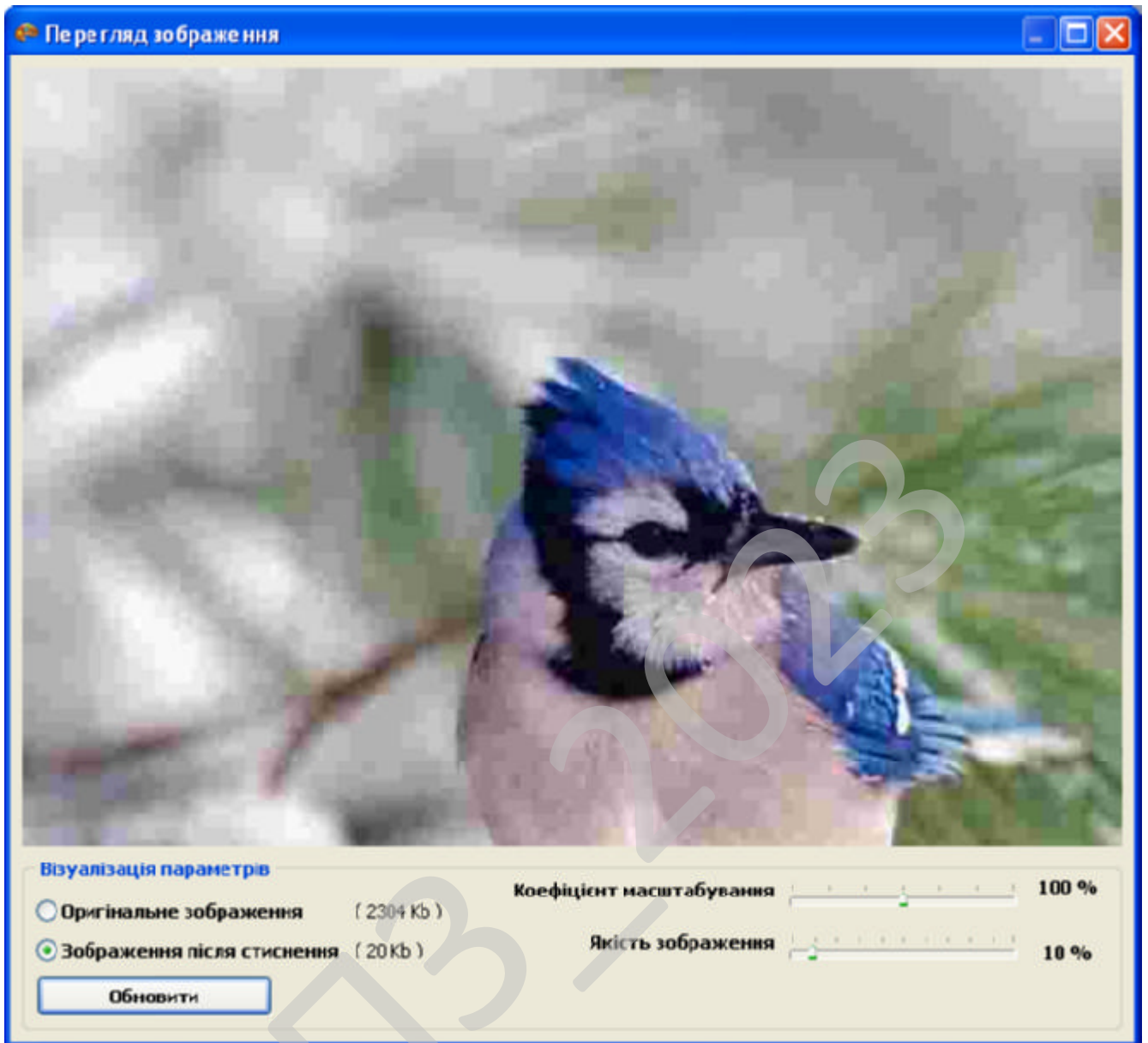


Рисунок 5.4 – Зображення після стиснення, якість зображення 10% (зменшення якості дуже помітне)

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		66

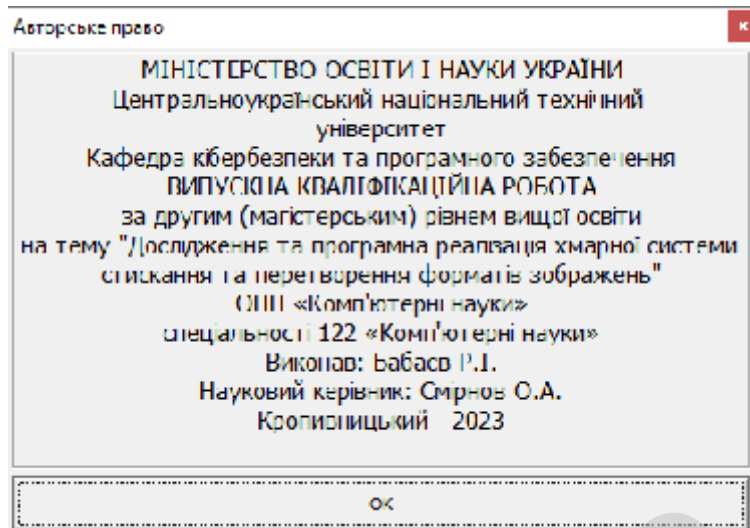


Рисунок 5.5 – Довідка про програму

На рисунку 5.5 наведена довідка про програму, у якій вказано, де виконувався магістерський проект, ким він виконувався й хто керівник студента.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		67

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для хмарної системи стискання та перетворення форматів зображень.

Метою розробки є дослідження та програмна реалізація хмарної системи стискання та перетворення форматів зображень.

Об'єктом дослідження є процес стискання та перетворення форматів зображень.

Предметом дослідження є методи стискання та перетворення форматів зображень.

Методи дослідження базуються на методах комп'ютерної графіки, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

- Удосконалено метод стискання та перетворення форматів зображень.
- Розроблено вітчизняний продукт стискання та перетворення форматів зображень, який має більш широкі можливості, на відміну від існуючих аналогів.

					VKPM-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		68

7 ДАНІ ПРО ЕКОНОМІЧНУ ЕФЕКТИВНІСТЬ РОЗРОБЛЕНОЇ ПРОГРАМИ

7.1 Техніко-економічне обґрунтування теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Після ознайомлення з підприємством та засобами розробки програмної продукції був розроблений план розробки програми. Був підрахований необхідний час для розробки та впровадження програми. Цей час склав 48 днів (два місяці). В магістерській роботі була досліджена та розроблена програмна реалізація хмарної системи стискування та перетворення форматів зображень.

Розроблене програмне забезпечення має достатню надійність і задовольняє усім поставленим умовам, а саме:

- а) невеликий розмір;
- б) невеликі системні потреби;
- в) незалежність від встановлених на комп'ютері баз даних;
- г) зручність у користуванні та надійність.

Таблиця 7.1 – Початкові дані

Показники	Позначення	Характеристика або величина
1	2	3
1. Кількість розроблених програм період, шт.	N	1
2. Кількість екземплярів програм, шт.	Ne	20
3. Запланований термін розробки, днів	Fpq	48 (2 місяці)
4. Група задачі підсистеми управління (1-6)	–	1
5. Ступінь новизни задачі (А, Б, В, Г)	–	Б
6. Складність алгоритму (1, 2, 3)	–	2

Продовження таблиці 7.1

1	2	3
7. Кількість макетів вхідної інформації	–	3
8. Кількість форм вихідної інформації.	–	4
9. Мова програмування (1-6)	–	2
10. Попередній досвід (1-6)	–	3
11. Гнучкість проекту ПП (1-6)	–	3
12. Детальність проекту ПП (1-6)	–	2
13. Рівень спрацьованості колективу (1-6)	–	2
14. Ступінь вимірності процесів (1-6)	–	3
15. Необхідна надійність програмного забезпечення (1-6)	–	2
16. Розмір бази даних (порівняно з розміром програми) (1-6)	–	2
17. Складність кінцевого програмного продукту (1-6)	–	2
18. Необхідний рівень забезпечення повторного використання (1-6)	–	2
19. Документованість відповідно до планованого життєвого циклу (1-6)	–	2
20. Вимоги до швидкодії ПП (1-6)	–	2
21. Обмеження на розміри основного сховища даних (1-6)	–	2
22. Різноманітність використовуваних обчислювальних платформ (1-6)	–	2
23. Професійний рівень аналітиків (1-6)	–	2
24. Професійний рівень програмістів (1-6)	–	2
25. Постійність складу команди розробників (1-6)	–	2
26. Досвід розробки додатків (1-6)	–	2
27. Досвід роботи з обчислювальною платформою (1-6)	–	2

Продовження таблиці 7.1

1	2	3
28. Досвід роботи з мовою і інструментами середовища розробки (1-6)	–	2
29. Досвід роботи з програмними інструментами розробки (1-6)	–	3
30. Розробка ПЗ для декількох серверів одночасно (1-6)	–	2
31. Вимоги до дотримання встановленого графіка робіт (1-6)	–	2
32. Вартість ПЗ у розробника (НМА), грн.	–	20000
33. Норматив додаткової зарплати, % :	Н _д	10
34. Норматив відрахувань у соціальні фонди, %	Н _с	22
35. Норматив загальногосподарських витрат, %	Н _г	15
36. Норматив витрат на освоєння нових мов програмування, %	Н _п	15
37. Рівень рентабельності програмної продукції, %	Р _е	50
38. Ставка податку на додану вартість, %	Н _{дв}	20

7.2 Розрахунок трудомісткості розробки програмної продукції

Значення трудомісткості розробки програмного забезпечення для стадій ТЗ, ЕК, ТП та ВП визначаємо по типовим нормам часу приведеним в додатках МВ. Стадія РП є найбільш тривалою і трудомісткою, що робить значний вплив на інші стадії проекту.

Визначимо трудомісткість розробки ПЗ для стадії РП.

Обчислюємо номінальні трудовитрати, люд-міс.:

$$T_{ном} = A \text{ Size}^B, \quad (7.1)$$

де: A – коефіцієнт Боема, $A = 2,45$;

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		71

Size – загальний об'єм відлагодженого програмного коду, тис. рядків;

B – показник ступеня, що визначається співвідношенням:

$$B = 1,01 + 0,001 \sum W_i, \quad (7.2)$$

де: W_i – сумарне значення п'яти показників (МВ, додаток 2), що відображають особливості розробки проекту програмного продукту (ПП) і колективу розробників.

$$B = 1,01 + 0,001(2,43 + 3,64 + 3,38 + 3,95 + 2,73) = 1,027.$$

$$T_{ном} = 2,45 \cdot 2,7^{1,026} = 6,78 \text{ люд-міс.}$$

Визначаємо уточнені (з урахуванням приведених в МВ додатку 3 сімнадцяти додаткових коефіцієнтів) трудовитрати, люд-міс.:

$$T_{уточн} = T_{ном} PV_j, \quad (7.3)$$

де: PV_j – добуток сімнадцяти додаткових коефіцієнтів, приведених в МВ додатку 3.

$$T_{уточн} = 6,78 \cdot (0,88 \cdot 0,93 \cdot 0,88 \cdot 0,91 \cdot 0,95 \cdot 1,1 \cdot 0,87 \cdot 1,22 \cdot 1,16 \cdot 1,1 \cdot 1,1 \cdot 1,12 \cdot 1,1 \cdot 1,1 \cdot 1,1) = 9,37 \text{ люд-міс.}$$

Ці коефіцієнти дозволяють диференційовано оцінювати результати роботи програмістів, беручи до уваги швидкодію програми, використання різноманітних обчислювальних платформ і інструментів розробки, взаємодію декількох серверів, вимоги до об'ємів баз даних і ін.

Визначаємо підсумкові трудовитрати по стадії робочий проект, люд-дні:

$$T_{РП} = 0,3 C T_{уточн}^{0,33 + 0,2(B-1,01)} S, \quad (7.4)$$

де: C – визначений емпірично коефіцієнт, запропонований авторами методики, (МВ, додаток 4);

S – коефіцієнт стиснення (або подовження) графіка робіт %, що дозволяє коректувати терміни розробки ПЗ згідно встановленим вимогам. Вибираємо в межах (25...350)%.

$$T_{РП} = 0,3 \cdot 2,66 \cdot 9,37^{0,33 + 0,2(1,026 - 1,01)} \cdot 100 = 168 \text{ люд/день.}$$

Для зручності визначення загальної трудомісткості на розробку програмного забезпечення результати розрахунків по стадіям зводимо до таблиці 7.2.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		72

Таблиця 7.2 – Визначення трудомісткості розробки програмного забезпечення

Стадії розробки	Трудомісткість за типовими нормами та розрахунками	
	Величина, люд/дні	Підстава
Технічне завдання	9	Д5
Ескізний проект	10	Д6
Технічний проект	9	Д7
Робочий проект	168	Ф 7.1-7.4
Впровадження	13	Д13
Всього	209	–

7.3 Визначення чисельності виконавців і планового фонду зарплати

Чисельність ставок інженерів-програмістів для розробки програмного забезпечення визначається за формулою:

$$Ч = \frac{T_{нз} \cdot N}{F_{pq} - H_{ев}}, \quad (7.5)$$

де: F_{pq} – плановий фонд робочого часу одного спеціаліста, днів;

$T_{нз}$ – трудомісткість розробки програмного забезпечення люд-дні.

$$Ч = \frac{209 \cdot 1}{60 - 5} = 3,8 \text{ ставки.}$$

Чисельність інженерів-електронщиків для проведення технічного обслуговування та ремонту комп'ютерних мереж визначається в залежності від наявності технічних засобів і норм витрат часу на виконання профілактичних робіт на протязі року.

Визначаємо затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за період розробки. Результати розрахунку зводимо до таблиці 7.3.

Таблиця 7.3 – Затрати часу на виконання профілактичних робіт по обслуговуванню обладнання за розрахунковий період

Найменування обладнання	Профілактичне обслуговування			
	Кількість хв. на один. обл.	Кількість обладнання	Затрати часу в хв.	Затрати часу в год.
Системний блок ПК	385	12	4620	77
Монітор	160	12	1920	32
Клавіатура	140	12	1680	28
Маніпулятор «мишка»	30	12	360	6
Принтер матричний	185	1	185	3
Принтер лазерний	355	2	710	12
Принтер струминний	300	1	300	5
Сканер	155	2	310	5
Концентратор– маршрутизатор	155	2	310	5
Кабельні господарства ЛОМ на 1 м.п.	2,5	100	250	4
Кабельне господарство електромережі	48	50	2400	40
Копіювальний апарат	285	2	570	10
Усього за рік:			3 _ч	227

Час на профілактику обладнання в загальному балансі робочого часу інженерів-електронщиків не повинен складати більше 10%.

Виходячи з цього фонд робочого часу інженерів-електронщиків складає:

$$\Phi_{op}^c = \frac{3_{ч} \cdot n_{mic}}{1,2}, \quad (7.6)$$

$$\Phi_{op}^c = \frac{227 \cdot 2}{1,2} = 378 \text{ год.}$$

Визначаємо необхідну кількість ставок штатного персоналу сектора ТО:

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		74

$$Ч_{ел} = \frac{\Phi_{др}^c}{F_{др} \cdot T_{зм}}, \quad (7.7)$$

$$Ч_{ел} = 378 / (48 \cdot 8) = 1 \text{ ставка.}$$

Для забезпечення нормального технічного обслуговування засобів ТО та мереж, необхідно прийняти найбільше ціле значення розрахункової чисельності інженерів-електронщиків.

Чисельність інженерів-системотехніків, адміністраторів мережі, дизайнерів WEB вузлів, системних програмістів (аналітиків), бухгалтерів-економістів визначається за потребою в залежності від функціональних обов'язків. Після визначення чисельності персоналу складається штатний розклад.

Таблиця 7.4 – Розрахунок чисельності штатного персоналу сектору системного та адміністративного обслуговування засобів ОТ та комп'ютерних мереж

Посада	Вид роботи	Час	К-ть штатних одиниць
Адміністратор загальної мережі, аналітик	Адміністрування локальної мережі, поштового та серверу DNS (OC FreeBSD), маршрутизатора Cisco, доменного контролеру Windows Server 2019, серверу доступу ADSL (OC Linux), налаштування ADSL, VPN PPPoE, Frame Relay, Wi-Fi	2	0,5
	Налаштування і конфігурування базової станції безпроводного зв'язку (CMTS)	0,5	
	Розробка та впровадження проектів з організації зв'язку між віддаленими об'єктами, ЛОМ	0,5	
	Забезпечення цілодобової роботи зв'язку клієнтів до мережі Інтернет	1	
Всього		4	

Продовження таблиці 7.4

Посада	Вид роботи	Час	К-ть штатних одиниць
Продакт-менеджер	Презентації нової продукції, пошук каналів збуту	1	0,25
	Підтримка постійних клієнтів	0,5	
	Оформлення договорів, ведення тендерів	0,25	
	Контроль взаєморозрахунків з постачальниками	0,25	
Всього		2	
Дизайнер WEB	Розробка концепції оформлення та інтерфейсу сайту, оптимізація дизайну існуючих, проектує їх структуру та навігацію	1	0,25
	Створення графічних і стилістичних елементів сайту	0,5	
	Оформлення банерів і промо-сторінок	0,25	
	Розміщення графіки і контенту на Інтернет сторінках	0,25	
Всього		2	
Інженер верстальник	Розробка та верстка макетів рекламної продукції та технічної документації	1	0,25
	Верстка друкованих видань	0,5	
	Додрукова підготовка макетів	0,25	
	Розміщення графіки і контенту на Інтернет сторінках	0,25	
Всього		2	

Складемо штатний розклад виконавців.

Таблиця 7.5 – Штатний розклад виконавців

Посада	Кількість ставок	Середньомісячний оклад, грн.	Всього за період розробки, грн.
Керівник (ІТ-менеджер)	1	13000	39000
Продакт-менеджер	0,25	12000	9000
Інженер-програміст	3,8	12000	136800
Інженер-електронщик	1,0	10500	31500
Інженер-системотехнік	0,25	10500	7875
Адміністратор мережі	0,5	10500	15750
Системний програміст	0,25	10500	7875
Дизайнер WEB	0,25	12000	9000
Інженер-верстальник	0,25	11700	8775
Бухгалтер-економіст	0,5	12500	18750
Всього за період розробки	$R_{cn} = 8,25$	-	$\Phi_{роб} = 284325$

Розрахуємо середньоденну зарплату одного виконавця:

$$z_{cd} = \frac{\Phi_{роб}}{R_{cn} F_{pq}}, \quad (7.8)$$

де: $\Phi_{роб}$ – загальна сума зарплати за плановий період, грн.

$$z_{cd} = \frac{284325}{8,25 \cdot 48} = 718 \text{ грн.}$$

7.4 Розрахунок капітальних вкладень та амортизаційних відрахувань у розробника

Балансова вартість будівель визначається з урахуванням кількості робочих місць виконавців, питомої площі на одне робоче місце, та вартості одного квадратного метра виробничої площі:

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		77

$$B_{y\partial} = R_{cn}^1 S_y C_{nl}, \quad (7.9)$$

де: R_{cn}^1 – кількість робочих місць виконавців, шт. Приймаємо 8 робочих місць;

S_y – питома площа на одне робоче місце, m^2 ;

C_{nl} – вартість одного квадратного метра площі, грн.

Згідно даних ТОВ науково-дослідницького консалтингового підприємства «Пектораль» (м. Кіровоград) ціна одного квадратного метра площі новобудови, вік якої не перевищує 25 років, по місту складає 500...1600 у.о./ m^2 . Враховуючи, що курс складає 1 у.о. = 37 грн. приймаємо для розрахунку вартість одного метра квадратного рівною 20000 грн./ m^2 . На кожне робоче місце у середньому потрібно 8 m^2 . З урахуванням цього:

$$B_{y\partial} = 8 \cdot 8 \cdot 20000 = 1280000 \text{ грн.}$$

Вартість передавальних пристроїв складає 10% від вартості будівель, і у даному випадку вона складе: 128000 грн.

Балансова вартість інвентарю розраховується за нормою 3500 грн. на одне робоче місце. Тобто:

$$I_{нв} = R_{cn}^1 \cdot C_m, \quad (7.10)$$

де: C_m – ціна меблів для одного робочого місця, грн.

$$I_{нв} = 8 \cdot 3500 = 28000 \text{ грн.}$$

Балансова вартість обчислювальної техніки визначається по оптовим цінам постачальника з врахуванням витрат на транспортування.

Специфікація на обчислювальну техніку наведена в таблиці 7.7.

Дані по оптовій ціні на обладнання та комплектуючі вибирались по прайсу Інтернет-магазину Компбест за 16.10.23 – джерело <https://compbest.com.ua>.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		78

Таблиця 7.6 – Специфікація

Найменування комплектуючої або обладнання	Тип	Оптова ціна
Персональний комп'ютер		10947
Системний блок		7347
Процесор	AMD Ryzen 3 4100 (100-100000510BOX) AM4, 4 ядра, 8 потоків, 3.8 GHz, 4.0 GHz TDP – 65 Вт, 7nm	–
Системна плата	GIGABYTE A520M H AM4, DDR4, LAN – Гб/с, DVI, HDMI, 1 x M.2 2280, 4 x SATA 6. Gb/s, Micro-ATX	–
Відеокарта	Radeon R7 350 2Gb Afox	–
Жорсткий диск	SSD M.2 2280 240GB Apacer 240 GB, TLC M.2, SATA III (6Gb/s)	–
Оперативна пам'ять	DDR4 8GB 2666 MHz Kingston	–
DVD-привод	-	–
Корпус	ATX Middle Tower FOXCONN Pro, 3GTLA 489, PSU 350W(FSP Brand: ATX-350PNR 12cm), black, (front bezel – black+light silver body material – 0.6mm), 80mm fan (rear 2xUSB2.0/AUDIO/MIC, Air Duct, Tool-less chassis design,Thermally Advantaged Chassis	–
Кулер	–	–
Кардрідер внутрішній	USB 2.0 Card reader STORM CR-35U1A4-E int. 3.5", 1*USB2.0+AUDIO+1394, multi: A Type Cards, black	–

Продовження таблиці 7.6

Найменування комплектуючої або обладнання	Тип	Оптова ціна
інше	Клавіатура, мишка	Подарунок
Монітор	22" TFT, ASUS VW223D (5ms, 300/3000: 170/160, D-SUB, Wide)	3600
Принтер лазерний	Canon i-SENSYS LBP6030W	2700
Принтер струминний	Epson Stylus Photo P50 (C11CA45341) + USB cable	5500
Копіювальний апарат	Canon i-SENSYS MF217W with Wi-Fi	5965

Витрати на транспорт, монтаж та випробування можуть бути прийняті в межах до 10% від оптової ціни.

Для визначення необхідної кількості капітальних вкладень складемо таблицю 7.8.

Таблиця 7.7 – Балансова вартість обчислювальної техніки

Найменування обчислювальної техніки	Кількість, шт.	Ціна за одиницю, грн.	Витрати на транспортування, монтаж та випробування.	Загальна вартість, грн.
Персональні комп'ютери	15	10947	16420,5	180625,5
Принтер лаз.	2	2700	540	5940
Принтер струм.	1	5500	550	6050
Копіюв. апарат (МФУ)	1	5965	596,5	6561,5
Всього	—	—	—	199177

Таблиця 7.8 – Вартість основних фондів та амортизаційні відрахування розробника

Групи та види основних фондів	Балансова вартість, грн.	Амортизація	
		Норма, %	Відрахування, грн.
1	2	3	4
Група 3			
1. Будівлі	1280000	-	-
2. Передавальні пристрої	128000	-	-
Всього по групі	1408000	5	70400
Група 4			
3. Обчислювальна техніка	199177	-	-
Всього по групі	199177	50	99588,5
Нематеріальні активи			
4. Нематеріальні активи	20000	10	2000
Група 5, 6			
5. Вимірювальні пристрої	9031	25	2257,75
6. Транспортні засоби	143000	20	28600
7. Господарський інвентар	28000	25	7000
Всього по групі	180031	-	5000
Разом	$K_p = 1807208$		$A_p = 176988,5$

Примітка: вартість автомобіля Sens (Standard+) взята по даним з автосалону «Кіровоград-Авто», джерело <http://kirovograd-avto.ukravto.ua/catalog/tm-9/model-80/description>, складає 143000 грн.

7.5 Визначення собівартості розробки та ціни програмної продукції

Визначимо основну зарплату виконавців:

$$Z_o = \frac{Z_{cd} \cdot T_{nz}}{N_e}, \quad (7.11)$$

де: N_e – кількість екземплярів програм, шт.

$$Z_o = 718 \cdot 209 / 20 = 7503 \text{ грн.}$$

Визначимо додаткову зарплату (оплата відпусток, виконання державних та суспільних обов'язків) на рівні 10%:

$$Z_d = Z_o \cdot H_q \cdot 0,01, \quad (7.12)$$

де: H_q – норматив додаткової зарплати, %.

$$Z_d = 7503 \cdot 10 \cdot 0,01 = 750,3 \text{ грн.}$$

Відрахування на соціальні потреби за нормативом $H_c = 22\%$ від суми основної та додаткової зарплати:

$$C_{oc} = 0,01 \cdot H_c (Z_o + Z_d), \quad (7.13)$$

де: H_c – відрахування на соціальні потреби, %.

$$C_{oc} = 0,01 \cdot 22(7503 + 750,3) = 3054 \text{ грн.}$$

Визначимо загальногосподарські витрати (електроенергію, ремонт і утримання приміщень і т.д) за нормативом $H_z = 15\%$ від основної зарплати:

$$G_{ocn} = Z_o \cdot H_z \cdot 0,01, \quad (7.14)$$

де: H_z – загальногосподарські витрати, %.

$$G_{ocn} = 7503 \cdot 15 \cdot 0,01 = 1125 \text{ грн.}$$

Визначимо витрати на матеріали для розробки програмної продукції за нормами споживання та діючими цінами за одиницю виміру:

$$Z_M = (Z_{M1} + Z_{M2} + Z_{M3}) / N_e, \quad (7.15)$$

де: Z_{M1} – вартість паперу, грн.; Z_{M2} – вартість запам'ятовуючих пристроїв, грн.; Z_{M3} – вартість фарби, картриджів, тонеру, грн.; N_e – кількість екземплярів програм, шт.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		82

Згідно виданих норм приймаємо 0,25 пачки паперу на місяць розробки. Тоді, враховуючи, що вартість пачки паперу складає $C_n = 210$ грн., визначаємо вартість паперу за період розробки $N_m = 2$ міс:

$$Z_{M1} = C_n \cdot N_m. \quad (7.16)$$

$$Z_{M1} = 210 \cdot 2 \cdot 0,25 = 105 \text{ грн.}$$

Згідно виданих норм до вартості запам'ятовуючих пристроїв входить вартість CD/DVD дисків в кількості 4 примірників.

$$Z_{M2} = \sum C_d, \quad (7.17)$$

де: C_d – вартість дисків CD/DVD: CDR TDK 700Mb, 80Min, 52x Cake box – 28 грн./шт., DVD-R LG 4,7Gb, 16x speed Cake box – 28 грн./шт.

$$Z_{M2} = 28 \cdot 4 = 112 \text{ грн.}$$

Згідно виданих норм одноразовій заправці підлягають усі друкуючі пристрої і становить:

$$Z_{M3} = \sum C_z, \quad (7.18)$$

де: C_z – вартість розхідних матеріалів друкуючих пристроїв: відновлення та заправка картриджу для Canon i-SENSYS LBP6030W – 574 грн.; картридж для Epson Stylus Photo P50 – 558 грн.; відновлення картриджу для MF217W – 570 грн.

$$Z_{M3} = 574 + 558 + 570 = 1702 \text{ грн.}$$

$$Z_M = (105 + 112 + 1702) / 20 = 96 \text{ грн.}$$

Визначимо витрати на освоєння нових мов програмування або операційних систем за нормативом ($H_n = 15\%$) від основної зарплати виконавців:

$$O_n = Z_o \cdot H_n \cdot 0,01, \quad (7.19)$$

де: H_n – норматив витрат на освоєння нових мов програмування, %.

$$O_n = 7503 \cdot 15 \cdot 0,01 = 1125 \text{ грн.}$$

Визначимо витрати на амортизацію основних фондів з урахуванням загальної річної суми амортизаційних відрахувань та кількості екземплярів програм ($N_e = 20$ прим.):

$$A_m = \frac{A_p \cdot N_{mic}}{N_e \cdot 12}, \quad (7.20)$$

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		83

де: A_p – загальна річна сума амортизаційних відрахувань, грн.

$$A_m = 176989 \cdot 2 / (20 \cdot 12) = 1475 \text{ грн.}$$

Повна собівартість ПЗ визначається як сума витрат за попередніми статтями калькуляції:

$$C_n = Z_o + Z_d + C_{oc} + \Gamma_{ocn} + Z_m + O_n + A_m. \quad (7.21)$$

$$C_n = 7305 + 730,5 + 3054 + 1125 + 96 + 1125 + 1475 = 14910,5 \text{ грн.}$$

Величини ціна підприємства, податок на додану вартість, відпускна ціна програмної продукції визначаються за формулами, приведеними в таблиці 7.9

Таблиця 7.9 – Нормативна калькуляція собівартості розробки програмного забезпечення задачі

Найменування статей витрат	Позначення	Величина, грн
1	2	3
1. Основна зарплата виконавців	Z_o	7305
2. Додаткова зарплата виконавців	Z_d	730,5
3. Відрахування на соціальні потреби	C_{oc}	3054
4. Загальногосподарські витрати	Γ_{ocn}	1125
5. Витрати на матеріали	Z_m	96
6. Освоєння нових операційних систем, мов програмування	O_n	1125
7. Амортизація основних фондів	A_m	1475
8. Повна собівартість програмного забезпечення	C_n	14910,5
9. Плановий прибуток	P_p	7455,25
10. Ціна підприємства $C_n = C_n + P_p$	C_n	22365,75
11. Податок на додану вартість $ПДВ = 0.01 \cdot N_{dv} \cdot C_n$	$ПДВ$	4473,15
12. Відпускна ціна програмної продукції $C = C_n + ПДВ$	C	26838,9

Визначимо плановий прибуток за рівнем рентабельності (P_n) програмної продукції, яка залежить від складності програми та ступеня новизни задачі.

Для даного програмного забезпечення рівень рентабельності складає 50%.

$$P_p = 0,01 \cdot P_n \cdot C_n, \quad (7.22)$$

де: P_n – рівень рентабельності, %.

$$P_p = 0,01 \cdot 50 \cdot 14910,5 = 7455,25 \text{ грн.}$$

7.6 Визначення об'єму капітальних вкладень у споживача програмної продукції

Об'єм капітальних вкладень у споживача програмної продукції визначаємо на основі балансової вартості основних фондів, яка враховує ціну, транспортно-заготівельні витрати, вартість будівель, монтажних та пусконаладжувальних робіт, а також витрати на випробування у виробничих умовах. Результати розрахунків зводимо у таблицю 7.10.

Таблиця 7.10 – Розрахунок об'єму капітальних вкладень у споживача програмної продукції

Найменування капітальних вкладень	Сума за варіантами, грн.	
	Базовий	Новий
Вартість програмної продукції	–	26839
Всього капітальних витрат	–	26839

7.7 Визначення експлуатаційних витрат

Експлуатаційні витрати у споживача програмної продукції визначаємо при умові роботи підсистеми на протязі року. Результати зводимо до таблиці 7.11.

Таблиця 7.11 – Розрахунок експлуатаційних витрат у споживача програмної продукції

Найменування статей витрат	Позначення	Сума витрат за варіантами, грн.	
		Базовий	Новий
1. Витрати на обслуговування	Z_p	59048	14762
2. Витрати на електроенергію	$Z_{ел}$	0	0
3. Витрати на амортизацію	$Z_{ам}$	0	6710
Всього витрат за рік	I	59048	21472

Витрати на профілактичні роботи:

$$Z_p = T_p \cdot Z_z \cdot (1 + 0,01 \cdot H_q) \cdot (1 + 0,01 \cdot H_c), \quad (7.23)$$

де: T_p – кількість годин обслуговування сервера за рік, год.;

Z_z – заробітна плата обслуговуючого персоналу, грн/год.

Після купівлі нового програмного забезпечення кількість профілактичних годин робіт зменшилось з 400 годин на рік до 100 годин на рік, тому витрати на технічне обслуговування зменшилися з:

$$Z_{p \text{ баз}} = 400 \cdot 110 \cdot 1,1 \cdot 1,22 = 59048 \text{ грн},$$

до:

$$Z_{p \text{ нов}} = 100 \cdot 110 \cdot 1,1 \cdot 1,22 = 14762 \text{ грн}.$$

Витрати на електроенергію визначаються з урахуванням спожитої потужності ($P_{ел}$) в кіловатах, часу експлуатації технічних засобів (T_p) в годинах та ціни однієї кіловат-години ($C_{ел}$):

$$Z_{ел} = P_{ел} \cdot T_p \cdot C_{ел}. \quad (7.24)$$

Визначити різницю споживання електроенергії при впровадженні систем не має можливості, тому витрати на електроенергію в розрахунку приймаємо рівними нулю.

Таблиця 7.13 – Показники економічної ефективності програмної продукції

Найменування показників	Одиниця виміру	Величина
1. Кількість екземплярів програми	Прим.	20
2. Повна собівартість розробленої програми	Грн.	14910
3. Ціна розробленої програми	Грн.	22365
4. Плановий прибуток від реалізації розробленої програми	Грн.	7455
5. Рентабельність програмної продукції	%	50
6. Об'єм додаткових капітальних вкладень у виробника програмної продукції	Грн.	1807208
7. Загальний прибуток від реалізації програмної продукції	Грн.	149100
8. Величина економічного ефекту при виготовлені програмної продукції	Грн.	104286
9. Період окупності додаткових капітальних вкладень у виробника програмної продукції	Років	2
10. Об'єм додаткових капітальних вкладень у споживача програмної продукції	Грн.	26839
11. Величина економічного ефекту у користувача програмної продукції	Грн.	30866
12. Період окупності додаткових капітальних вкладень у користувача програмної продукції	Роки	0,7

Визначимо величину економічного ефекту у користувача програмної продукції за формулою:

$$E_{cn} = (I_{\delta} - I_n) - E_n(K_n - K_{\delta}), \quad (7.27)$$

де: I_{δ} , I_n – величина експлуатаційних витрат за базовим и новим варіантом відповідно;

K_{δ} , K_n – об'єм капітальних вкладень за варіантами, що порівнюються.

$$E_{cn} = (59048 - 21472) \cdot 0,25 \cdot 26839 = 30866 \text{ грн.}$$

Визначимо період окупності додаткових капітальних вкладень у споживача програмної продукції за рахунок зниження експлуатаційних витрат:

$$T_{cn} = \frac{K_n - K_0}{I_0 - I_n}, \quad (7.28)$$

$$T_{cn} = \frac{26839}{59048 - 21472} = 0,7 \text{ роки.}$$

Показники економічної ефективності програмної продукції зводимо до таблиці 7.13.

7.9 Висновки

Розроблена програма економічно вигідна. За рахунок впровадження програмного забезпечення досягається скорочення часу обробки інформації, підвищується культура праці, підвищення якості приймаючих управлінських рішень.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		89

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Законом України “Про охорону праці” [3] регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин», та ДСанПіН 3.3.2-007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин».

Програмісти у процесі роботи мають негативний вплив на органи зору, а також мають значну розумову напругою і нервово-емоційне навантаження. Руки (суглоби пальців та м'язи рук) при роботі з клавіатурою мають теж істотне навантаженням [1]. До шкідливих факторів, які впливають на робітників галузі інформаційних технологій (ІТ) спеціалісти відносять високочастотні електромагнітні коливання (випромінювання) роботи апаратної частини ЕОМ та виділення шкідливих газів.

Ці шкідливі фактори можуть привести до професійних захворювань.

При розгляді шкідливих чинників роботи програмістів та інших спеціалістів ІТ будемо керуватись наступними нормативно-правовими актами: «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98, та «Правила охорони праці під час експлуатації електронно-обчислювальних машин» НПАОП 0.00-1.28-10,

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		90

Умови праці програміста вилучають наступні фактори:

- параметри повітряного середовища в приміщенні;
- вентиляція приміщення;
- освітлення приміщення;
- параметри повітряного середовища в приміщенні, тощо.

Щоб запропонувати заходи щодо зменшення впливу комп'ютера на організм програміста визначемо фактори, які можуть викликати професійне захворювання і впливають на працездатність програміста,

8.2 Шкідливі і небезпечні фактори при роботі з комп'ютером

Електронно-обчислювальні машини (ЕОМ) та інше обладнання є джерелами небезпеки ураження електричним струмом.

На робочому місці програміста виникають небезпечні та шкідливі фактори: підвищений рівень шуму, несприятливі мікрокліматичні умови, недостатній рівень освітленості, шкідливі речовини, підвищений рівень електромагнітних випромінювань радіочастот, висока напруга електричної мережі, статична електрика та інші.

При роботі з використанням ЕОМ відзначають наступні небезпечні та шкідливі фактори:

- ризик виникнення надзвичайних ситуацій природного або штучного характеру на об'єкті або території.
- ризик виникнення пожежі;
- негативний вплив на органи зору людини;
- ризики ураження електричним струмом;
- недостатня, або надмірна освітленість робочого місця;
- електромагнітні (у т.ч. високочастотні) електромагнітні випромінювання (коливання);
- несприятливі мікрокліматичні умови;

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		91

- нервово-емоційна напруженість праці;
- інтелектуальні навантаження;
- монотонність праці;
- невідповідність ергономічних показників робочого місця діючим вимогам;
- шум;
- статичні навантаження на кістково-м'язовий апарат.

Вивчення умов праці на робочому місці програміста є необхідною умовою запобігання впливу небезпечних та шкідливих факторів на організм персоналу.

Дослідження санітарно-гігієнічних умов у приміщенні звичайно проводять з використанням пристроїв вимірювання параметрів мікроклімату – вологи повітря (психрометр), освітленості (люксметр) та шуму (шумомір), а також відомих розрахункових методик.

8.3 Пожежна безпека

Пожежна безпека на підприємстві іт-індустрії являє собою комплекс організаційних заходів щодо забезпечення нормального протипожежного стану об'єкта нерухомості – житлового, виробничого, складського, офісного або торгово-розважальної споруди, приміщення або комплексу приміщень.

Нормативною основою цього питання на підприємствах усіх форм власності є Кодекс цивільного захисту України та його 13-й розділ, а також Правила пожежної безпеки України. Згідно цих нормативно-правових актів керівник і власник підприємства, компанії або організації несе повну адміністративну та кримінальну відповідальність за своєчасне і правильне введення і підтримання протипожежного режиму. Згідно Державного стандарту України 2272:2006 «Пожежна безпека. Терміни та визначення основних понять» [8] під протипожежним режимом “розуміється звід правил поведінки людей,

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		92

виконання робіт і експлуатації об'єкта нерухомості, призначений для забезпечення пожежної безпеки”.

Керівник в свою чергу може делегувати частину своїх обов'язків і повноважень спеціально призначеному наказом по підприємству співробітнику. У більшості випадків призначається одна відповідальна посадова особа по підприємству або компанії в цілому і окремо – в кожному з окремих будівель, філій, відділів, дивізійних і структурних підрозділів.

Призначений співробітник приступає до виконання своїх обов'язків на підставі наказу по підприємству. Відповідальну особу в обов'язковому порядку знайомлять з цим документом, який він запевняє своїм підписом. У кожному з окремих приміщень, що охороняються повинна бути розташована інформаційна бирка з прізвищем, ім'ям, по батькові та номером контактної телефону особи, відповідальної за протипожежну безпеку. Конкретний перелік обов'язків щодо забезпечення режиму на об'єкті, комплектації, збереження та експлуатації засобів та інструментів протипожежного захисту вказується в посадових інструкціях, положеннях і наказах.

Чинними нормами і законодавчими актами України не передбачений точний список вимог, що висуваються до відповідальної особи – вік, гендерна приналежність та інші фактори значення не мають. Найчастіше компанії та організації висувають на цю посаду висококваліфікованого фахівця з вищою або середньою технічною професійною освітою, що має безперервний стаж роботи в галузі діяльності підприємства не менше ніж три роки. Крім того, бажано, що б здобувач мав мінімальний досвід діяльності в сфері пожежної безпеки, деякі технічні вміння та навички, Досвід роботи з комп'ютером. Відповідальна особа повинна пройти навчання з питань пожежної безпеки з відривом від виробництва, а в подальшому проходити курси в навчальному центрі не рідше, ніж один раз на три роки.

До кола обов'язків особи, відповідальної за протипожежну безпеку в компанії, організації або на підприємстві входять:

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		93

– розробка і ведення документації – планів з евакуації персоналу в разі пожежі, ведення журналів обліку, наказів, і спеціальних інструкцій щодо захисту об’єктів і використання первинних засобів гасіння, інструкцій з техніки безпеки, технічного обслуговування вогнегасників різних марок і типів, систем пожежної сигналізації та інше;

– організація навчання працівників, проведення інструктажів;

– матеріально-технічне забезпечення підприємства протипожежним інвентарем – первинними засобами пожежогасіння, включаючи переносні і пересувні вогнегасники, інформаційними стендами з актуальними матеріалами, пожежними щитами і знаками пожежної безпеки та іншими;

– забезпечення технічної справності протипожежного обладнання, інструментів та спеціального інвентарю;

– періодичні перевірки стану ввіреного об’єкта;

– здача звітності до контролюючих органів, допомога у здійсненні планових та позапланових перевірок;

– організація безпечної евакуації співробітників і персоналу, матеріальних цінностей, організація гасіння вогнищ загоряння або задимлення в разі виникнення пожежі.

– Крім евакуаційного плану в будівлях і приміщеннях з масовим скупченням людей повинні бути інструкції з евакуації. Плани та інструкції повинні бути розвішені в приміщеннях на добре видимих місцях. Не рідше, ніж один раз на календарний рік на підприємствах повинні проводитися тренування на випадок виникнення надзвичайних ситуацій.

8.4 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		94

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень;

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог.

Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язково наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга).

Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

8.5 Розрахункова частина

Для захисного штучного заземлення застосовуються вертикальні електроди: метелевий куток $80 \cdot 50 \cdot 6$ мм., (згідно з ДСТУ 8769:2018 «Кутики сталеві гарячекатані нерівнополичні. Сортамент») довжиною $L=1,7$ м., та горизонтальний електрод – металева полоса з перетином $60 \cdot 5$ мм. Напруга – $220/380$ В. Розрахункова схема розташування заземлюючих електродів – у ряд.

Розрахунок проведемо за допустимим опором розтіканню струму заземлювача.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		95

Визначаємо необхідну кількість вертикальних електродів заземлювача (без вхарування горизонтального заземлювача), при $R_{3H} = 4 \text{ Ом}$:

$$N = R_0 / (K_{ев} R_{3H}) = 21,9 / (0,8 \cdot 4) = 6,85 \approx 7 \text{ шт.}$$

Визначаємо довжину з'єднуючої полоси:

$$L_{\Pi} = 1,05 \cdot A \cdot N = 1,05 \cdot 1,7 \cdot 7 = 12,22 \approx 12 \text{ м.}$$

Опір розтіканню електричного струму з'єднуючої полоси з урахуванням кліматичного коефіцієнта питомого опору ґрунта K_{Π} [10]:

$$R_{\Pi} = 0,366(\rho \cdot K_{\Pi} / L_{\Pi}) \lg(2 \cdot L_{\Pi}^2 / (B \cdot t)) = \\ = 0,366(40 \cdot 5 / 12) \cdot \lg((2 \cdot 12^2) / (0,06 \cdot 0,6)) = 23,5 \text{ Ом.}$$

де $K_{\Pi} = 5$ – табличне значення кліматичного коефіцієнта питомого опору ґрунта для відповідної кліматичної зони для з'єднуючої полоси [10]:

$B = 60 \text{ мм.} = 0,06 \text{ м.}$ – ширина з'єднуючої полоси (задана).

Загальний опір розтіканню електричного струму заземлювача [10]:

$$R = (R_0 \cdot R_{\Pi}) / (R_0 \cdot \eta_{\Pi} + N \cdot R_{\Pi} \cdot K_{ев}) = \\ = (21,9 \cdot 23,5) / (21,9 \cdot 0,75 + 9 \cdot 23,5 \cdot 0,8) = 3,48 \text{ Ом.}$$

де $\eta_{\Pi} = 0,75$ – табличне значення коефіцієнта екранування з'єднуючої полоси [10].

Умова $R \leq R_{3H}$ виконується ($3,48 \leq 4$).

Так як при 7 вертикальних електродах R суттєво менше R_{3H} , зменшимо кількість вертикальних електродів N до 6 і виконаємо перерахунок. У результаті остаточно отримали: $R = 3,98 \text{ Ом.}$ при кількості вертикальних електродів $N = 6$.

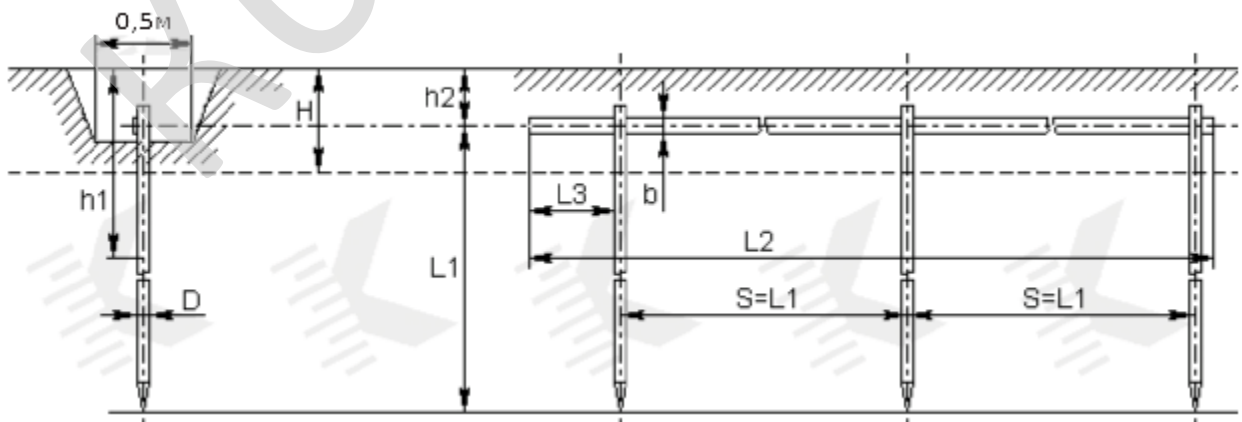


Рисунок 8.1 – Схема штучного заземлення

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		97

7. Оришака, О. В. Основи охорони праці: навч. посіб. / О. В. Оришака, Г. П. Горбачова, К. М. Марченко; М-во освіти і науки України, Центральноукраїн. нац. техн. ун-т. – Кропивницький : ЦНТУ, 2022. – 175 с. – Режим доступу до ресурсу: <http://dspace.kntu.kr.ua/jspui/handle/123456789/12161> (дата звернення 19.09.23).

8. Оришака О.В. Охорона праці в галузі та цивільний захист / О.В Оришака, Г.П. Горбачова, О.М. Мезенцева, К.М. Марченко, К.О. Буравченко; М-во освіти і науки України, Центральноукраїн. нац. техн. ун-т. – Кропивницький: Видавець Лисенко В.Ф., 2019. – 226 с. – Режим доступу до ресурсу: <http://dspace.kntu.kr.ua/jspui/handle/123456789/9258> (дата звернення 19.09.23).

9. Методичні рекомендації до виконання розділу "Заходи з охорони праці та техніки безпеки" випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти для здобувачів вищої освіти спеціальностей 123 "Комп'ютерна інженерія" та 122 "Комп'ютерні науки" / М-во освіти і науки України, Центральноукраїн. нац. техн. ун-т, каф. кібербезпеки та програм. забезпечення; [укл. О.В. Оришака, К.М. Марченко]. – Кропивницький: ЦНТУ, 2022. – 19 с. [Електронний ресурс]. – Режим доступу: <http://dspace.kntu.kr.ua/jspui/handle/123456789/12240> (дата звернення 19.09.23).

10. Охорона праці. Ч. 1. Захисне заземлення : метод. вказ. до викон. розрахунків з викор. персон. ЕОМ IBM сумісного типу / Кіровоград. ін-т с.-г. машинобуд. ; [укл. О. В. Оришака, Є. К. Солових, В. О. Оришака]. – Кіровоград : КІСМ, 1997. – 20 с. – Режим доступу: <http://dspace.kntu.kr.ua/jspui/handle/123456789/4358> (дата звернення 19.09.23).

11. Наказ Міністерства регіонального розвитку, будівництва та житлово-комунального господарства України 31.10.2016 «Про затвердження ДБН В.1.1-702016» – Режим доступу до ресурсу: <https://ips.ligazakon.net/document/fn025551> (дата звернення 19.09.23).

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		99

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для хмарної системи стискання та перетворення форматів зображень.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів стискання та перетворення форматів зображень.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем стискання та перетворення форматів зображень.

– Досліджена система стискання та перетворення форматів зображень.

– На основі отриманих результатів досліджень створена програмна реалізація хмарної системи стискання та перетворення форматів зображень.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання стискання та перетворення форматів зображень.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		100

При створенні програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Delphi 10.4 Sydney. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм ДСТУ 7564:2014.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Розроблена програма має реальний економічний ефект від її впровадження у виробництво у сумі 30866 грн. З урахуванням вартості розробки програми та обладнання, строк окуплення становить 0,7 роки.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		101

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бабаєв Р.І. Дослідження та програмна реалізація хмарної системи стискання та перетворення форматів зображень // Збірник праць молодих науковців ЦНТУ. – Вип. 14. – Кропивницький: ЦНТУ, 2023.
2. Alasdair McAndrew. A Computational Introduction to Digital Image Processing. Chapman & Hall. 2021. 560 p.
3. PeterShirley, SteveMarschner. Fundamentals of Computer Graphics. 2009
4. Михайло Пічугін, Іван Канкін, Володимир Воротніков Комп'ютерна графіка. Навчальний посібник / Центр навчальної літератури 346 с. 2019р.
5. Маценко В.Г. Комп'ютерна графіка: Навчальний посібник. – Чернівці: Рута, 2009 – 343 с.
6. Інженерна комп'ютерна графіка: підручник / В.В. Проців [та ін.] / М-во освіти і науки України, Нац. гірн. унт-т. – Дніпро: НГУ, 2017. – 247 с.
7. Проців В.В. Прикладна комп'ютерна графіка [Текст]: Навч. посібник / В.В. Проців, К.А. Зіборов, К.М. Бас, Г.К. Ванжа; М-во освіти і наук, Нац. гірн. унт. – Д.: НГУ, 2016. – 187 с.
8. Kopf, Johannes and Lischinski, Dani. Depixelizing Pixel Art (англ.) // ACM Trans. Graph.. – 2011. – Vol. 30, no. 4. – P. 99:1--99:8.
9. Giachetti, Andrea and Asuni, Nicola. Real-Time Artifact-Free Image Upscaling (англ.) // Trans. Img. Proc.. – 2011. – Vol. 20, no. 10. – P. 2760—2768.
10. Smirnov, O., Odarchenko, R., Smirnova, T., Bondar, S., Volosheniuk, D. «Optimal Structure Construction of Private 5G Network for the Needs of Enterprises». Lecture Notes on Data Engineering and Communications Technologies, 2023, 178, pp. 208–223.
11. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» CEUR Workshop Proceedings, Volume 3187, 2022,

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		102

12. Smirnov O., Smirnova T., Anas M. Al-Oraiqat, Drieiev O., Polishchuk L., Sheroz Khan, Yassin M. Y. Hasan, Aladdein M. Amro, Hazim S. AlRawashdeh «Method for Determining Treated Metal Surface Quality Using Computer Vision Technology». Sensors (Basel, Switzerland) Volume 22, Issue 16, 6223, 2022.

13. Smirnov O., Kuznetsov A., Kryvinska N., Kiian A., Kuznetsova K. «Full Non-Binary Constant-Weight Codes». SN Computer Science, Vol 2, 337, 2021. <https://doi.org/10.1007/s42979-021-00739-w>

14. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O. «Hiding Messages in Audio Files Using Direct Spread Spectrum». 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021, Cracow, Poland, 22-25 September 2021. P. 414-418.

15. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». 4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) – 2021, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

16. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». International Journal of Computing; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

17. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». CEUR Workshop Proceedings. Volume 2740, 2020, Pages 102-114.

18. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». Journal of theoretical and applied information technology Vol.98. No 21, 2020, P. 3334-3346.

19. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New technique for data hiding in cover images using adaptively generated pseudorandom sequences». CEUR Workshop Proceedings Volume 2654, 2020, Pages 1-14.

20. Smirnov O., Kuznetsov A., Onikiychuk A., Makushenko T., Anisimova O., Arischenko A. «Adaptive pseudo-random sequence generation for spread spectrum image steganography». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 161-165.

21. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT), Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

22. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

23. Smirnov, O., Drieieva, H., Drieiev, O., Polishchuk, Y., Brzhanov, R., Aleksander, M. «Method of fractal traffic generation by a model of generator on the graph». CEUR Workshop Proceedings Volume 2616, 2020, Pages 366-379.

24. Smirnov, O., Drieieva, H., Drieiev, O., Simakhin, V., Bondar, S., Odarchenko, R. «Managing multifractal properties of the binary sequence generated with the Markov chains», CEUR Workshop Proceedings Volume 2608, 2020, Pages 633-645.

25. Smirnov, O., Kuznetsov, A., Gorbacheva, L., Babenko, V., «Hiding data in images using a pseudo-random sequence», CEUR Workshop Proceedings Volume 2608, 2020, Pages 646-660.

26. Zhurakovskiy, B., Tsopa, N., Batrak, Y., Odarchenko, R., Smirnova, T «Comparative analysis of modern formats of lossy audio compression». Workshop Proceedings, 2020, 2654, стр. 315-327.

27. Smirnov O. Kuznetsov A., Zaichenko Yu., Pastukhov M., Oleshko O., Kuznetsova K., «Formation of Discrete Signals with Special Correlation Properties». International Conference on Information and Telecommunication Technologies and Radio Electronics, UkrMiCo 2019; Odessa; Ukraine; 9-13 September 2019. P.22-28.

28. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 – Research Institute for Intelligent Computer Systems – 2019. – P. 393-407.

29. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), Kyiv, Ukraine, 8 – 11 October 2019 . P.517-522.

30. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». CEUR Workshop Proceedings, Vol 2588, P. 90-106, 2019.

31. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Averchev, A., Pastukhov, M., Kuznetsova, K., «Formation of Pseudorandom Sequences with Special Correlation Properties», 2019 3rd International Conference on Advanced Information and Communications Technologies, AICT -2019/ Lviv, Ukraine, 2-6 July, 2019, P. 395-399.

32. Smirnov, O., Kuznetsov, A., Kavun, S., Babenko, B., Nakisko, O., Kuznetsova, K., «Malware Correlation Monitoring in Computer Networks of Promising Smart Grids», 2019 IEEE 6th International Conference On Energy Smart Systems (2019 IEEE ESS), Kyiv, Ukraine April 17-19, 2019 P. 347-352.

33. Smirnov, O., Kuznetsov, A., Kovalchuk, D., Pastukhov, M., Kuznetsova, K., Prokopovych-Tkachenko, D., «Discrete Signals with Special Correlation Properties», CEUR Workshop Proceedings Volume 2353, CEUR Workshop Proceedings 2019, Pages 618-629.

34. Smirnov A.A., Kuznetsov A.A., Danilenko D.A., Berezovsky A., «The statistical analysis of a network traffic for the intrusion detection and prevention systems», Telecommunications and Radio Engineering. – Volume 74, Issue 1. – Begel House Inc. – 2015. – P. 61-78.

35. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New Technique for Hiding Data in Cover Images Using Adaptively Generated Pseudorandom Sequences». CEUR Workshop Proceedings Volume 2732, 2020, Pages 214-227.

36. Т.В. Смірнова, О.М. Дреєв, О.А. Смірнов «Хмарна інформаційна система оцінювання шорсткості з використанням дискретного частотного аналізу макروفотografій». IV міжнародна науково-практична конференція «Інформаційна безпека та комп'ютерні технології», м. Кропивницький. 15-16 квітня 2021р. – Кропивницький: ЦНТУ. – 2021. – С. 30.

37. О.А. Смірнов, П.С. Усік, «Дослідження перспектив використання технологічних рішень в мережах 5G» у Кібербезпека та інформаційні технології: монографія. – Х. : ТОВ «ДІСА ПЛЮС», 2020.С. 122-135.

38. О.А.Смірнов, Т.В.Смірнова, Л.І. Поліщук, К.О. Буравченко, А.О.Макевнін, «Дослідження хмарних технологій як сервісів», Кібербезпека: освіта, наука, техніка. № 3(7). С. 43-62. 2020.

39. Смірнов О.А., Дреєва Г.М., Дреєв О.М., Смірнова Т.В. «Фрактальний аналіз генератора самоподібного трафіку на основі ланцюга Маркова». Центральнoукраїнський науковий вісник. Технічні науки. № 2(33). с. 161-172, 2019.

40. О. Смірнов, Є. Деменко, О. Онікійчук, А. Арищенко, Л. Горбачова, «Формування псевдовипадкових послідовностей для приховування даних в зображеннях» Комп'ютерні науки та кібербезпека. № 4. С. 30-37. 2019.

41. Смірнов О.А., Коноплицька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В. Поліщук Л.І. Проектування комп'ютерних систем та мереж. Навчальний посібник – Кропивницький: вид. Лисенко В.Ф. 2019. – 264 с.

42. Smirnov, O., Kuznetsov, A., Kuznetsova., K. Synthesis of Discrete Signals with Improved Correlation Properties. Монографія: In.: ISCI'2019: Information Security in Critical Infrastructures. Collective monograph. Edited by Ivan D. Gorbenko and Alexandr A. Kuznetsov, ASC Academic Publishing, USA, 2019, pp. 281-299. – ISBN: 978-0-9989826-8-7 (Hardback), ISBN: 978-0-9989826-9-4 (Ebook).

43. Смірнов О.А., Дреєва Г.М. Метод генерування фрактального трафіку за допомогою моделі генератора на графі. Монографія: Інформаційна безпека та інформаційні технології : монографія / за заг. ред. В. С. Пономаренка. – Х. : Вид. Рожко С.Г. 2019. С. 123-139

44. Дреєва Г.М., Смірнов О.А., Дреєв О.М. Метод генерування фрактальноподібної числової послідовності на основі скінченного автомату для моделювання трафіку у мережі. Центральноукраїнський науковий вісник. Технічні науки. № 1(32). с. 173-183, 2019.

45. Смірнов О.А., Кавун С.В., Коваленко О.В., Дреєв О.М. Мережні інформаційні технології. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 159 с.

46. Смірнов О.А., Смірнов С.А. Дідик А.К., Дреєв О.М. Моделі системи нейромережових експертів безпечної маршрутизації у хмарних антивірусних системах. Збірник наукових праць "Системи обробки інформації". – Випуск 3 (140). – Х.: ХУПС – 2016. – С. 36-39.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		107

47. Смірнов О.А., Кавун С.В., Коваленко О.В., Доренський О.П., Дреєв О.М., Вялкова В.І. Комп'ютерні мережі. Навчальний посібник – Кіровоград: РВЛ КНТУ, 2016. – 233 с.

48. Смірнов О.А., Дреєв О.М. Порівняння бітових щільностей при використанні різних методів кодування інформації. Збірник наукових праць "Системи обробки інформації". – Випуск 2 (118). т.2. – Х.: ХУПС – 2014. – С. 64-67

49. Смірнов О.А., Дреєв О.М. Порівняння бітових щільностей при використанні різних методів кодування інформації. Збірник тез VI міжнародної науково-практичної конференції "Проблеми та перспективи розвитку IT-індустрії". м. Харків. 17-18 квітня 2014р. – Харків: ХНЄУ. – 2014. – С. 240.

50. Смірнов О.А., Коваленко О.В., Кожанова А.С., Лешко О.Л., Константинова Л.В. Основи системного програмування. Навчальний посібник для студентів вищих навчальних закладів напрямів підготовки 8.050102 «Комп'ютерна інженерія». За ред. Коваленка О.В., Гриф "Навчальний посібник" надано у відповідності з листом Міністерства освіти і науки України від 26.02.2013 року № 1/11-4368. – Кіровоград: КНТУ 2013. – 257с.

51. Смірнов О.А., Дреєв О.М., Доренський О.П. «Дослідження впливу ступеня стиснення зображень на оперативність їх доставки у телекомунікаційній системі. Збірник наукових праць "Системи обробки інформації". – Випуск 8(115). – Х.: ХУПС – 2013. – С. 234-239.

52. Смірнов О.А., Доренський О.П., Дреєв О.М. Аналіз процесів стиснення та відновлення зображень на основі цифрових методів. Наука і техніка Повітряних сил Збройних Сил України. – Випуск 3(12). – Х.: ХУПС. – 2013. – С.122-127.

					ВКРМ-122.23.0027.00.00.ПЗ	Арк.
Вим.	Арк.	№ докум.	Підпис	Дата		108

Додаток А
(обов'язковий)

Технічне завдання

Зміст

1 Найменування та область застосування.....	2
2 Підстава для розробки.....	2
3 Мета та призначення розробки.....	2
4 Джерела розробки.....	2
5 Технічні вимоги.....	2
5.1 Вміст проекту.....	2
5.2 Показники призначення.....	3
5.3 Вимоги до функціональних характеристик.....	3
5.4 Вимоги до архітектури.....	3
5.5 Вимоги до надійності.....	3
5.6 Умови експлуатації.....	4
5.7 Вимоги до складу та параметрів технічних засобів.....	4
5.8 Вимоги до інформаційної і програмної сумісності.....	4
5.8.1 Обладнання.....	4
5.8.2 Мова програмування.....	4
5.8.3 Вхідні дані.....	5
5.8.4 Вихідні дані.....	5
6 Вимоги до програмної документації.....	5
7 Економічні вимоги.....	5
8 Вимоги щодо охорони праці.....	5
9 Перелік документів, що розробляються.....	6
10 Етапи розробки.....	6
11 Порядок контролю та приймання.....	6

					ВКРМ-122.23.0027.00.00.ТЗ		
Вим.	Арк.	№ документа	Підпис	Дата			
Розробив	Бабасєв Р.І.				Літ.	Аркуш	Аркушів
Перевірів	Смірнов О.А.						
Н. Контр.	Коваленко А.С.				ЦНТУ КН-22М-1		
Затв.	Смірнов О.А.						

1 Найменування та область застосування

Це технічне завдання розповсюджується на дослідження та програмну реалізацію хмарної системи стискання та перетворення форматів зображень.

2 Підстава для розробки

Підставою для розробки служить завдання на випускню кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, видане на кафедрі кібербезпеки та програмного забезпечення (нак. № 32-13 від 04.08.2023 року).

3 Мета та призначення розробки

Метою випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є дослідження та програмна реалізація хмарної системи стискання та перетворення форматів зображень.

4 Джерела розробки

Джерелом цієї випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти є стосовна до теми література і існуючі аналоги.

5 Технічні вимоги

5.1 Склад продукції

Складниками розробки є:

- вибір і обґрунтування методів реалізації проекту;
- розробка програмної частин системи, а також розробка взаємодії системи з ОС та з користувачем;

					ВКРМ-122.23.0027.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

- техніко-економічне обґрунтування доцільності прийнятого до розробки програмного забезпечення;
- аналіз умов праці;
- розробка програми, що реалізує спроектовані алгоритми роботи системи.

5.2 Показники призначення

Система повинна забезпечувати:

- програмну реалізацію хмарної системи стискання та перетворення форматів зображень;
- цілісність даних у процесі роботи та при зберіганні;
- простий, інтуїтивно зрозумілий інтерфейс.

5.3 Вимоги до функціональних характеристик

Розроблене програмне забезпечення не повинно мати обмежень на версію драйверів та операційної системи.

5.4 Вимоги до архітектури

Компонент, що розробляється повинен використовувати системні засоби та апаратні засоби, що на даному етапі розвитку обчислювальної техніки найбільше поширені.

5.5 Вимоги до надійності

Програмні модулі написані по всім правилам, які стосуються стандартних викликів процедур, функцій, методів і форм, визначених технічною документацією на середовище розробки.

					ВКРМ-122.23.0027.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		3

5.6 Умови експлуатації

Робочі місця користувачів ПЗ повинні задовольняти наступним умовам експлуатації:

- температура повітря: 19-20 град. по Цельсію;
- відносна вологість повітря до 80%;
- атмосферний тиск 107 кПа.

5.7 Вимоги до складу та параметрів технічних засобів

Програмне забезпечення повинно бути реалізоване на ПЕОМ архітектури IBM PC, працювати в ОС Windows 10/11 і з сумісними з цією платформою пристроями і прикладним програмним забезпеченням.

5.8 Вимоги до інформаційної і програмної сумісності

Переносність програмного забезпечення повинна бути забезпечена за рахунок його реалізації стандартного інтерфейсу взаємодії з ОС, що працюють під управлінням ОС Windows 10/11.

5.8.1 Обладнання

Комп'ютер Intel® Celeron/8 Mb/1.2 Gb/SVGA 14" 1Mb або сумісні з ним.

5.8.2 Мова програмування

Середовище Delphi 10.4 Sydney.

					ВКРМ-122.23.0027.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		2

5.8.3 Вхідні дані

Опис алгоритму роботи запропонованої системи.

5.8.4 Вихідні дані

Робоча програма.

6 Вимоги до програмної документації

Програмна продукція повинна бути представлена у виді опису структури даних, схем та опису алгоритму, а також текстів вихідних модулів програмного забезпечення згідно ЄСПД .

7 Економічні вимоги

7.1 Для ПЗ необхідно виробити функціонально-вартісний аналіз варіантів розробки.

7.2 Виконати розрахунок витрат показників економічного ефекту з урахуванням цін на 3 вересня 2023 року.

8 Вимоги щодо охорони праці

В частині охорони праці випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти повинна бути розглянута пожежна безпека.

					ВКРМ-122.23.0027.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		5

9 Перелік документів, що розробляються

- Наукова новизна – 1 аркуш.
- Структурна схема системи – 1 аркуш.
- Функціональна схема системи – 1 аркуш.
- Діаграма процесів – 1 аркуш.
- Блок-схема алгоритму роботи програми – 2 аркуша.
- Показники економічної ефективності – 1 аркуш.
- Пояснювальна записка – 108 аркушів.

10 Етапи розробки

10.1 Збір і обробка інформації по темі випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти. Постановка задачі на виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти (складання ТЗ).

10.2 Проведення досліджень або експериментальних робіт для уточнення основних положень випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти.

10.3 Розробка функціональних схем, блок схем алгоритмів роботи програмного забезпечення.

10.4 Побудова схем взаємодії даних.

10.5 Створення прототипу ПЗ.

10.6 Віднаходження ПЗ, аналіз отриманих результатів.

10.7 Робота над питанням охорони праці і техніки безпеки.

10.8 Розрахунок з техніко-економічного обґрунтування.

10.9 Оформлення пояснювальної записки і виконання робіт по графічній частині.

11 Порядок контролю та приймання

11.1 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на попередній захист 10.12.2023 р.

11.2 Подання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти на захист 19.12.2023 р.

					ВКРМ-122.23.0027.00.00.ТЗ	Арк.
Вим.	Арк.	№ документа	Підпис	Дата		6

Додаток Б
(обов'язковий)

Міністерство освіти і науки України
Центральноукраїнський національний технічний університет

ЗАТВЕРДЖУЮ

Керівник випускної кваліфікаційної роботи за
другим (магістерським) рівнем вищої освіти
_____ Смірнов О.А.

*Дослідження та програмна реалізація
хмарної системи стискання та перетворення форматів зображень*

Лістинг програми

Код документу 12

Носій: CD/DVD-диск / USB-флеш-накопичувач

Загальна кількість аркушів: 28

Літера: РП

Кропивницький – 2023 року

EasyCompressAlgoritm_stysku.dpr - файл проекту

```

program EasyCompressAlgoritm_stysku;

uses
  Windows, SysUtils, Classes, Forms, Messages,
  Main in 'Main.pas' {MainForm}, // головна програма
  Preview in 'Preview.pas' {DisplayForm}, // підпрограма перегляду зображень
  Error in 'Error.pas' {ErrorForm}, // підпрограма виявлення помилок
  About in 'About.pas' {AboutForm}, // підпрограма виведення даних про розробника
  SfxUtils in 'TZIP\SfxUtils.pas', // підпрограма центру роботи з зображеннями
  Zip in 'TZIP\Zip.pas', // підпрограма стиснення зображення
  ZipDlls in 'TZIP\ZipDlls.pas'; // бібліотека стиснення зображення

var
  ClassName : Array[0..255] of char;
  WM_PARAM_ATOM : integer;
  result : integer;
  h: THandle;
  Param_Atom: Atom;
  nb : integer;

{$R EasyCompressAlgoritm_stysku.res}

begin
  {
    Application.Initialize;
    // Application.Title := 'Starting EasyCompressAlgoritm_stysku...';
    GetClassName(Application.Handle, ClassName, 254);
    // result := FindWindow(ClassName, 'EasyCompressAlgoritm_stysku ');
    h := FindWindow(nil, 'EasyCompressAlgoritm_stysku ');

    if result <> 0 then begin
      WM_PARAM_ATOM := RegisterWindowMessage('WM_PARAM_ATOM');
      if ParamCount > 0 then begin
        for nb := 1 to ParamCount do begin
          Param_Atom := GlobalAddAtom(PChar(ParamStr(nb)));
          SendMessage(h, WM_PARAM_ATOM, Param_Atom, 0);
        end;
      end;
      ShowWindow(result, SW_RESTORE);
      SetForegroundWindow(result);
      Application.Terminate;
    end else begin
      Application.Title := '';
      // Application.Icon.Assign("chinaz23.ico");
      Application.CreateForm(TMainForm, MainForm);
      Application.CreateForm(TErrorForm, ErrorForm);
      Application.CreateForm(TAboutForm, AboutForm);
      Application.CreateForm(TDisplayForm, DisplayForm);
      Application.Run;
    end;
  }
end.

```

Main.pas - основна програма

```

unit Main;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ComCtrls, StdCtrls, ExtCtrls, Buttons, ShellCtrls, XPMan, Types,
  ShellAPI, FileCtrl, Math, Jpeg, OleCtrls, StrUtils, DateUtils, Zip, SfxUtils,
  IniFiles;
type
  TMainForm = class(TForm)
    Panel_Parameters: TGroupBox;
    Panel_Files: TGroupBox;
    Panel_Compression: TGroupBox;
    Panel_Destination: TGroupBox;
    Btn_CloseApp: TBitBtn;
    Btn_AboutApp: TBitBtn;
    Btn_BeginCompression: TBitBtn;
    Btn_AddFile: TBitBtn;
    Btn_DeleteFile: TBitBtn;
    Btn_ViewFile: TBitBtn;
    CompressionRate: TTrackBar;
    RBtn_KeepOGSize: TRadioButton;
    RBtn_ChangeOGSize: TRadioButton;
    RBtn_ModifiedSize: TRadioGroup;
    RBtn_OverwriteFiles: TRadioButton;
    RBtn_CreateNewFolder: TRadioButton;
    RBtn_CreateZIPFile: TRadioButton;
    CBox_CreateBackup: TCheckBox;
    RBtn_KeepAllMetadata: TRadioButton;
    RBtn_RemoveAllMetadata: TRadioButton;
    CBox_ChangeDateData: TCheckBox;
    FilesSelectedDate: TDateTimePicker;
    CBox_AddCopyright: TCheckBox;
    CopyrightText: TEdit;
    Btn_ChangeCopyrightTextFormat: TBitBtn;
    PBar_ProgressionStatus: TProgressBar;
    Label_Title1: TLabel;
    Label_Title2: TLabel;
    Label_Advice1: TLabel;
    Label_Advice2: TLabel;
    Label_CompressionRate: TLabel;
    Label_ProgressionStatus: TLabel;
    SaveZIPFileDialog: TSaveDialog;
    CopyrightPositionTL: TShape;
    CopyrightPositionTR: TShape;
    CopyrightPositionBL: TShape;
    CopyrightPositionBR: TShape;
    XPManifest: TXPManifest;
    Label_Title3: TLabel;
    AddImageDialog: TOpenDialog;
    FileList: TListBox;
    BackupPath: TEdit;
    CopyToDirectoryPath: TEdit;
    ExportZipFilePath: TEdit;
    Btn_BrowseBackupPath: TBitBtn;
    Btn_BrowseCopyToDirectoryPath: TBitBtn;
    Btn_BrowseExportZipFilePath: TBitBtn;
    CopyrightFontDialog: TFontDialog;
    FilesSelectedTime: TDateTimePicker;
    CBox_DeleteThumbnails: TCheckBox;
    CBox_AddGlobalComment: TCheckBox;
    CommentText: TEdit;
    Labell: TLabel;
    Imagem: TImage;
  end;
end;

```

```

Timer1: TTimer;
procedure RBtn_ModifiedSizeClick(Sender: TObject);
procedure FormShow(Sender: TObject);
procedure Btn_AboutAppClick(Sender: TObject);
procedure CBox_AddGlobalCommentClick(Sender: TObject);
procedure Btn_BeginCompressionClick(Sender: TObject);
procedure Btn_ChangeCopyrightTextFormatClick(Sender: TObject);
procedure SelectedProfileChange(Sender: TObject);
procedure Btn_BrowseExportZipFilePathClick(Sender: TObject);
procedure Btn_BrowseCopyToDirectoryPathClick(Sender: TObject);
procedure Btn_BrowseBackupPathClick(Sender: TObject);
procedure CopyrightPositionMouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure CBox_AddCopyrightClick(Sender: TObject);
procedure RBtn_MetadataChoiceClick(Sender: TObject);
procedure RBtn_DestinationPathChoiceClick(Sender: TObject);
procedure RBtn_OGSizeChoiceClick(Sender: TObject);
procedure CompressionRateChange(Sender: TObject);
procedure Btn_CloseAppClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure Btn_ViewFileClick(Sender: TObject);
procedure Btn_DeleteFileClick(Sender: TObject);
procedure Btn_AddFileClick(Sender: TObject);
procedure ChangeControlsDisponibility(EnableControls: boolean);
procedure FileListDrawItem(Control: TWinControl; Index: Integer; Rect:
TRect; State: TOwnerDrawState);
procedure FileListKeyUp(Sender: TObject; var Key: Word; Shift: TShiftState);
procedure FileListClick(Sender: TObject);
procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);
procedure AddFilesFromPath(Path: string);
procedure FilesSelectedTimeChange(Sender: TObject);
procedure LoadProfiles();
procedure ReadProfile(ProfileIndex: integer);
procedure Timer1Timer(Sender: TObject);
private
  { Private Declarations }
  ZipComponent : TZip;
public
  { Public Declarations }
  procedure DefaultHandler(var msg); override;
end;

var
  MainForm: TMainForm;

  WM_PARAM_ATOM: cardinal;
  ErrorsList: TStringList;
  JPEGImage1: TJpegImage;
  JPEGImage2: TJpegImage;
  //      \
  //      /      BMPTempImage1: TBitmap;
  BMPTempImage2: TBitmap;
  FIRST_EXECUTION: boolean = true;
  PROFILE_APPLYING: boolean = false;
  false; //      \      IS_PROCESSING_IMAGES: boolean =
  ABORT_REQUESTED: boolean = false;
  APP_PATH: string;
  EXIFTOOL_PATH: string;
  TEMP_PATH: string;
  PROFILES_CFGFILE: string;

implementation

uses Error, About, Preview;

{$R *.dfm}

```

```

//Функція запуску та чекання завантаження

function LaunchAndWait(CommandLine: String; WShowWin: Word): Boolean;
var
  StartInfo    : TStartupInfo;
  ProcessInfo  : TProcessInformation;
  Ended        : Boolean;
begin
  FillChar(StartInfo, SizeOf(StartInfo), #0);
  StartInfo.cb := SizeOf(StartInfo);
  StartInfo.dwFlags:=STARTF_USESHOWWINDOW;
  StartInfo.wShowWindow:=WShowWin;
  if CreateProcess(nil, PChar(CommandLine), nil, nil, False, 0, nil, nil,
  StartInfo, ProcessInfo) then begin
    Ended := False;
    repeat
      case WaitForSingleObject(ProcessInfo.hProcess, 200) of
        WAIT_OBJECT_0: Ended := True;
        WAIT_TIMEOUT  : ;
      end;
      Application.ProcessMessages;
    until Ended;
    result := true;
  end else result := false;
end;

// ----- //
//Функція видалення директорії
// ----- //
function DeleteDirectory(Path: string): Boolean;
var
  fos: TSHFileOpStruct;
begin
  Path := ExcludeTrailingPathDelimiter(Path);
  ZeroMemory(@fos, SizeOf(fos));
  with fos do begin
    wFunc := FO_DELETE;
    fFlags := FOF_SILENT or FOF_NOCONFIRMATION;
    pFrom := PChar(Path + #0);
  end;
  Result := (0=ShFileOperation(fos));
end;

// ----- //
Зміна дати та часу файлу
// ----- //
function CorruptFileDateTime(const FileName: string; NewDate: TDateTime):
boolean;
var
  fHandle : integer;
  Succeed : boolean;
  FinalDate, TempFileTime : TFileTime;
  TempSystemTime : TSystemTime;
begin
  fHandle := FileOpen(FileName, fmShareDenyWrite or fmOpenWrite);
  if fHandle < 0 then Succeed := false
  else begin
    DecodeDateTime(NewDate, TempSystemTime.wYear, TempSystemTime.wMonth,
    TempSystemTime.wDay,
    TempSystemTime.wHour, TempSystemTime.wMinute,
    TempSystemTime.wSecond, TempSystemTime.wMilliseconds);
    SystemTimeToFileTime(TempSystemTime, TempFileTime);
    LocalFileTimeToFileTime(TempFileTime, FinalDate);
    Succeed := SetFileTime(fHandle, @FinalDate, @FinalDate, @FinalDate);
    FileClose(fHandle);
  end;
  Result := Succeed;
end;

```

```

// ----- //
Визначення імені файлу
// ----- //
function ExtractFileNameOnly(FileName:TFileName): TFileName;
var
  ExtensionPart : TFileName;
  ExtensionLength : Integer;
begin
  FileName := ExtractFileName(FileName);
  ExtensionPart := ExtractFileExt(FileName);
  ExtensionLength := Length(ExtensionPart);
  Delete(FileName, Length(FileName)-ExtensionLength+1,ExtensionLength);
  Result:=FileName;
end;
// ----- //
Пошук імені доступного файлу
// ----- //
function FindAvailableFileName(Directory, FileName, FileExtension: string):
string;
var
  FileDuplicationChars : string;
  FileDuplicationIndex : integer;
begin
  Directory := IncludeTrailingPathDelimiter(Directory);
  if FileExists(Directory + FileName + FileExtension) then begin
    FileDuplicationIndex := 2;
    FileDuplicationChars := ' ('+IntToStr(FileDuplicationIndex)+)';
    while FileExists(Directory + FileName + FileDuplicationChars +
FileExtension) do begin
      Inc(FileDuplicationIndex);
      FileDuplicationChars := ' ('+IntToStr(FileDuplicationIndex)+)';
    end;
    end else FileDuplicationChars := '';

    result := Directory + FileName + FileDuplicationChars + FileExtension;
end;
// ----- //
Створення рядка
// ----- //
function RandomString(GeneratedStringLength: integer): string;
var
  i: integer;
  BaseChars, TempStr: string;
begin
  BaseChars := 'abcdefghijklmnopqrstuvwxyz1234567890ABCDEFGHIJKLMNPOQRSTUVWXYZ';
  // Available Chars (they need to be compatible with a FileName)
  for i:=0 to GeneratedStringLength-1 do TempStr := TempStr +
BaseChars[Random(61)+1];
  result := TempStr;
end;

// Procedures Concerning Buttons Management
procedure TMainForm.Btn_AddFileClick(Sender: TObject);
var
  i: integer;
begin
  if AddImageDialog.Execute then begin
    ErrorsList := TStringList.Create;
    ErrorsList.Clear;
    for i:=0 to AddImageDialog.Files.Count-1 do begin
      if FileExists(AddImageDialog.Files[i]) then
FileList.Items.Add(AddImageDialog.Files[i]);
    end;
    if (ErrorsList.Text<>'') then begin
      ErrorForm.Caption := 'Помилка при додаванні файлів';
      ErrorForm.ErrorTitleMsg.Caption := 'Ці файли не можуть бути додані в список
зображень для стиснення '+ #13 +', тому що їх формат не підтримується
(підтримуються формати ALGORITHM_STYSKU, JPEG, BMP)';
      ErrorForm.FilesConcernedMemo.Lines := ErrorsList;
    end;
  end;
end;

```

```

        ErrorForm.ShowModal;
    end;
    ErrorsList.Free;
end;
end;
// ----- //
Додавання шляху до файлу
// ----- //
procedure TMainForm.AddFilesFromPath(Path: string);
var
    SearchInfo: TSearchRec;
begin
    if FileExists(Path) then begin
        if (UpperCase(ExtractFileExt(Path)) = '.ALGORITM_STYSKU')
            or (UpperCase(ExtractFileExt(Path)) = '.JPEG')
            or (UpperCase(ExtractFileExt(Path)) = '.BMP')
            or (UpperCase(ExtractFileExt(Path)) = '.SHM')
            then FileList.Items.Add(Path)
            else ErrorsList.Add(Path);
        exit;
    end;
    Path := IncludeTrailingPathDelimiter(Path);
    if DirectoryExists(Path) then begin
        if FindFirst(Path+'*.*', faAnyFile, SearchInfo)=0 then begin
            repeat
                if (SearchInfo.Name[1]<>'.') then
                    AddFilesFromPath(Path+SearchInfo.FindData.cFileName);
            until FindNext(SearchInfo)<>0;
            FindClose(SearchInfo);
        end;
    end;
end;
// ----- //
Знищення файлу
// ----- //
procedure TMainForm.Btn_DeleteFileClick(Sender: TObject);
begin
    FileList.DeleteSelected;
    Btn_DeleteFile.Enabled := false;
    Btn_ViewFile.Enabled := false;
    FileList.ItemIndex:=-1;
end;
// ----- //
Перегляд файлу
// ----- //
procedure TMainForm.Btn_ViewFileClick(Sender: TObject);
begin
    // Preview of the focused image
    if FileList.ItemIndex = -1 then exit;
    DisplayForm.LoadPreview(FileList.Items[FileList.ItemIndex]);
end;
//=====//
// Виведення списку файлів та вибір файлу
// ----- //
procedure TMainForm.FileListClick(Sender: TObject);
begin
    Btn_DeleteFile.Enabled := (FileList.SelCount <> 0);
    Btn_ViewFile.Enabled := (FileList.SelCount <> 0);
end;
// ----- //
Поле списку файлів
// ----- //
procedure TMainForm.FileListDrawItem(Control: TWinControl; Index: Integer; Rect:
TRect; State: TOwnerDrawState);
begin
    if odSelected in State then begin
        FileList.Canvas.Brush.Color := clInfoBk;
        FileList.Canvas.FillRect(Rect);
    end else begin

```

```

    FileList.Canvas.Brush.Color := clWhite;
    FileList.Canvas.FillRect(Rect);
end;
FileList.Canvas.Font.Name := 'Tahoma';
FileList.Canvas.Font.Size := 8;
FileList.Canvas.Font.Color := clblack;
FileList.Canvas.TextOut(Rect.Left+2, Rect.Top,
ExtractFileName(FileList.Items[Index]));
end;
// ----- //
procedure TMainForm.FileListKeyUp(Sender: TObject; var Key: Word; Shift:
TShiftState);
begin
    if (Key=VK_DELETE) then Btn_DeleteFileClick(nil);
    if (Key=VK_INSERT) then Btn_AddFileClick(nil);
    if (Key=VK_RETURN) then Btn_ViewFileClick(nil);
end;
// ----- //
Робота з файлами зображень
// ----- //
procedure TMainForm.DefaultHandler(var msg);
var
    ReceivedParam: PChar;
    ReceivedAtom: atom;
    params: TStringList;
    i, NumberOfFiles: integer;
    StrFileName : string;
    FileName : array[0..255] of char;
begin
    inherited DefaultHandler(Msg);
    if IS_PROCESSING_IMAGES then exit;
    if TMessage(msg).Msg = WM_PARAM_ATOM then begin
        ReceivedAtom := TMessage(msg).wParam;
        GetMem(ReceivedParam, 256);
        try
            GlobalGetAtomName(ReceivedAtom, ReceivedParam, 256);
            try
                params := TStringList.Create;
                ErrorsList := TStringList.Create;
                try
                    params.Clear;
                    params.Add(ReceivedParam);
                    ErrorsList.Clear;
                    for i := 0 to params.Count-1 do AddFilesFromPath(params[i]);
                    if (ErrorsList.Text<>'') then begin
                        ErrorForm.Caption := 'Помилка при додаванні файлів';
                        ErrorForm.ErrorTitleMsg.Caption := 'Ці файли не можуть бути додані в
список зображень для стиснення '+ #13 '+, тому що їх формат не підтримується
(підтримуються формати ALGORITHM_STYSKU, JPEG, BMP):';
                        ErrorForm.FilesConcernedMemo.Lines := ErrorsList;
                        ErrorForm.ShowModal;
                    end;
                    finally ErrorsList.Free; params.Free; end;
                    finally GlobalDeleteAtom(ReceivedAtom); end;
                    finally FreeMem(ReceivedParam); end;
                end;
            end;
        // ----- . .
        // Перетаскування файлів
        if TMessage(msg).Msg=WM_DROPFILES then begin
            NumberOfFiles := DragQueryFile(TMessage(msg).wParam, $FFFFFFFF, FileName,
sizeof(FileName));
            ErrorsList := TStringList.Create;
            ErrorsList.Clear;
            for i := 0 to NumberOfFiles-1 do
                begin
                    DragQueryFile(TMessage(msg).wParam, i, FileName, sizeof(FileName));
                    StrFileName := FileName;
                    AddFilesFromPath(StrFileName); // Calling for recursive addition procedure
                end;
        end;
    end;
end;

```

```

    if (ErrorsList.Text<>') then begin
        ErrorForm.Caption := 'Помилка при додаванні файлів';
        ErrorForm.ErrorTitleMsg.Caption := 'Ці файли не можуть бути додані в
        список зображень для стиснення '+ #13 +', тому що їх формат не підтримується
        (підтримуються формати ALGORITHM_STYSKU, JPEG, BMP):';
        ErrorForm.FilesConcernedMemo.Lines := ErrorsList;
        ErrorForm.ShowModal;
    end;
    ErrorsList.Free;
end;
end;
// ----- //
Виведення вікна
// ----- //
procedure TMainForm.FormShow(Sender: TObject);
var
    i: integer;
begin
    if not FIRST_EXECUTION then exit
    else FIRST_EXECUTION := false;
    WM_PARAM_ATOM := RegisterWindowMessage('WM_PARAM_ATOM');
    if ParamCount = 0 then exit;
    ErrorsList := TStringList.Create;
    try
        ErrorsList.Clear;
        for i := 1 to ParamCount do AddFilesFromPath(ParamStr(i));    if
        (ErrorsList.Text<>') then begin
            ErrorForm.Caption := 'Помилка при додаванні файлів';
            ErrorForm.ErrorTitleMsg.Caption := 'Ці файли не можуть бути додані в
            список зображень для стиснення '+ #13 +', тому що їх формат не підтримується
            (підтримуються формати ALGORITHM_STYSKU, JPEG, BMP):';
            ErrorForm.FilesConcernedMemo.Lines := ErrorsList;
            ErrorForm.ShowModal;
        end;
        finally ErrorsList.Free; end;
    end;
    procedure TMainForm.CompressionRateChange(Sender: TObject);
    begin

        DisplayForm.CompressionRate.Position:=MainForm.CompressionRate.Position;

        Label_CompressionRate.Caption := IntToStr(CompressionRate.Position)+' %';
    end;
    // ----- //
    Вибір коефіцієнту стискання
    // ----- //
    procedure TMainForm.RBtn_OGSizeChoiceClick(Sender: TObject);
    begin
        RBtn_ModifiedSize.Enabled := RBtn_ChangeOGSize.Checked;
    end;
    // ----- //
    procedure TMainForm.RBtn_ModifiedSizeClick(Sender: TObject);
    begin
        // if not PROFILE APPLYING then SelectedProfile.ItemIndex := 0;
    end; //=====//
    procedure TMainForm.RBtn_DestinationPathChoiceClick(Sender: TObject);
    begin
        // if not PROFILE APPLYING then SelectedProfile.ItemIndex := 0;
        CBox_CreateBackup.Enabled := RBtn_OverwriteFiles.Checked;
        BackupPath.Enabled := CBox_CreateBackup.Checked and
        RBtn_OverwriteFiles.Checked;
        Btn_BrowseBackupPath.Enabled := CBox_CreateBackup.Checked and
        RBtn_OverwriteFiles.Checked;
        CopyToDirectoryPath.Enabled := RBtn_CreateNewFolder.Checked;
        Btn_BrowseCopyToDirectoryPath.Enabled := RBtn_CreateNewFolder.Checked;
        ExportZipFilePath.Enabled := RBtn_CreateZIPFile.Checked;
        Btn_BrowseExportZipFilePath.Enabled := RBtn_CreateZIPFile.Checked;
    end;
end;

```

```

// ----- //
Вибір місця розташування резервної копії
// ----- //
procedure TMainForm.Btn_BrowseBackupPathClick(Sender: TObject);
var
  SelectedPath: string;
begin
  SelectedPath := BackupPath.Text;
  {$IFDEF VER170}
  if not SelectDirectory('Місце розташування резервної копії', '', SelectedPath,
[sdNewUI, sdNewFolder, sdShowEdit, sdValidateDir, sdShowShares], nil) then exit;
  {$ELSE}
  if not SelectDirectory('Місце розташування резервної копії', '', SelectedPath)
then exit;
  {$ENDIF}
  BackupPath.Text := IncludeTrailingPathDelimiter(SelectedPath);
end;
// ----- //
Місце розташування зображення
// ----- //
procedure TMainForm.Btn_BrowseCopyToDirectoryPathClick(Sender: TObject);
var
  SelectedPath: string;
begin
  SelectedPath := CopyToDirectoryPath.Text;
  {$IFDEF VER170}
  if not SelectDirectory('Місце розташування зображення', '', SelectedPath,
[sdNewUI, sdNewFolder, sdShowEdit, sdValidateDir, sdShowShares], nil) then exit;
  {$ELSE}
  if not SelectDirectory('Місце розташування зображення', '', SelectedPath) then
exit;
  {$ENDIF}
  CopyToDirectoryPath.Text := IncludeTrailingPathDelimiter(SelectedPath);
end;
// ----- //
procedure TMainForm.Btn_BrowseExportZipFilePathClick(Sender: TObject);
begin
  SaveZIPFileDialog.FileName := ExportZipFilePath.Text;
  if not SaveZIPFileDialog.Execute then exit;
  ExportZipFilePath.Text := SaveZIPFileDialog.FileName;
end;
//=====//
procedure TMainForm.RBtn_MetadadataChoiceClick(Sender: TObject);
begin
// if not PROFILE APPLYING then SelectedProfile.ItemIndex := 0;
  CBox_ChangeDateData.Enabled := RBtn_RemoveAllMetadadata.Checked;
  CBox_DeleteThumbnails.Enabled := RBtn_RemoveAllMetadadata.Checked;
  FilesSelectedDate.Enabled := CBox_ChangeDateData.Checked and
RBtn_RemoveAllMetadadata.Checked;
  FilesSelectedTime.Enabled := CBox_ChangeDateData.Checked and
RBtn_RemoveAllMetadadata.Checked;
end;
// ----- //
procedure TMainForm.CBox_AddCopyrightClick(Sender: TObject);
begin
// if not PROFILE APPLYING then SelectedProfile.ItemIndex := 0;
  CopyrightText.Enabled := CBox_AddCopyright.Checked;
  Btn_ChangeCopyrightTextFormat.Enabled := CBox_AddCopyright.Checked;

  if CBox_AddCopyright.Checked then begin

    Label_Title3.Font.Color := clBlack;

    if CopyrightPositionTL.Tag=0 then begin
      CopyrightPositionTL.Cursor := crHandPoint; CopyrightPositionTL.Brush.Color
:= clWhite;
    end else begin
      CopyrightPositionTL.Cursor := crArrow; CopyrightPositionTL.Brush.Color :=
clRed;

```

```

end;

if CopyrightPositionBL.Tag=0 then begin
    CopyrightPositionBL.Cursor := crHandPoint; CopyrightPositionBL.Brush.Color
:= clWhite;
end else begin
    CopyrightPositionBL.Cursor := crArrow; CopyrightPositionBL.Brush.Color :=
clRed;
end;

if CopyrightPositionTR.Tag=0 then begin
    CopyrightPositionTR.Cursor := crHandPoint; CopyrightPositionTR.Brush.Color
:= clWhite;
end else begin
    CopyrightPositionTR.Cursor := crArrow; CopyrightPositionTR.Brush.Color :=
clRed;
end;

if CopyrightPositionBR.Tag=0 then begin
    CopyrightPositionBR.Cursor := crHandPoint; CopyrightPositionBR.Brush.Color
:= clWhite;
end else begin
    CopyrightPositionBR.Cursor := crArrow; CopyrightPositionBR.Brush.Color :=
clRed;
end;

end else begin

    Label_Title3.Font.Color := clMedGray;
    CopyrightPositionTL.Brush.Color := clBtnFace;
    CopyrightPositionTR.Brush.Color := clBtnFace;
    CopyrightPositionBL.Brush.Color := clBtnFace;
    CopyrightPositionBR.Brush.Color := clBtnFace;

    CopyrightPositionTL.Cursor := crArrow; CopyrightPositionTR.Cursor :=
crArrow;
    CopyrightPositionBL.Cursor := crArrow; CopyrightPositionBR.Cursor :=
crArrow;

end;
end;
// ----- //
procedure TMainForm.CopyrightPositionMouseUp(Sender: TObject; Button:
TMouseButton; Shift: TShiftState; X, Y: Integer);
begin
// if not PROFILE APPLYING then SelectedProfile.ItemIndex := 0;
if not CBox_AddCopyright.Checked then exit;
CopyrightPositionTL.Tag := 0; CopyrightPositionTR.Tag := 0;
CopyrightPositionBL.Tag := 0; CopyrightPositionBR.Tag := 0;
(Sender as TShape).Tag := 1;
CBox_AddCopyrightClick(nil);
end;
// ----- //
procedure TMainForm.CBox_AddGlobalCommentClick(Sender: TObject);
begin
// if not PROFILE APPLYING then SelectedProfile.ItemIndex := 0;
CommentText.Enabled := CBox_AddGlobalComment.Checked;
end;
// ----- //
procedure TMainForm.Btn_ChangeCopyrightTextFormatClick(Sender: TObject);
begin
CopyrightFontDialog.Execute;
end;
// ----- //
procedure TMainForm.FilesSelectedTimeChange(Sender: TObject);
begin
FilesSelectedDate.Time := FilesSelectedTime.Time;
end;
//=====//

```

```

// Profiles Management
procedure TMainForm.LoadProfiles;
var
  NumberOfProfiles, DefaultIndex, i: integer;
  TempProfileName: string;
begin
  // SelectedProfile.Items.Text := 'Налагодження...';
  if not FileExists(PROFILES_CFGFILE) then begin
    exit;
  end;

  with TIniFile.Create(PROFILES_CFGFILE) do try
    if (ReadString('HEADER','ID','error') <> '17D5B2702D39F1A6C1E3ACEDAE99CC45')
  then exit;
    NumberOfProfiles := ReadInteger('HEADER','NumberOfProfiles_int',0);
    if NumberOfProfiles <= 0 then begin
      // SelectedProfile.ItemIndex := 0;
      exit;
    end;
    for i:=1 to NumberOfProfiles do begin
      TempProfileName := ReadString('PROFILE'+IntToStr(i),'Name_str','');
      if TempProfileName='' then TempProfileName:='Невідомий профіль';
      // SelectedProfile.Items.Add(TempProfileName);
    end;
    // Default ПроЗавантаження файлу
    DefaultIndex := ReadInteger('HEADER','DefaultProfile_int',0);
    // if (DefaultIndex>0) and (DefaultIndex<SelectedProfile.Items.Count) then
    SelectedProfile.ItemIndex := DefaultIndex
    // else SelectedProfile.ItemIndex := 0;
    // ReadProfile(SelectedProfile.ItemIndex);

  finally
    Free;
  end;
end;
// ----- //
procedure TMainForm.ReadProfile(ProfileIndex: integer);
var
  ProfileID: string;
  TempRate: integer;
begin
  if (ProfileIndex=-1) or (ProfileIndex=0) then exit;
  if not FileExists(PROFILES_CFGFILE) then begin
    LoadProfiles;
    exit;
  end;
  with TIniFile.Create(PROFILES_CFGFILE) do try
    if ReadString('HEADER','ID','error') <> '17D5B2702D39F1A6C1E3ACEDAE99CC45'
  then exit;
    ProfileID := 'PROFILE'+IntToStr(ProfileIndex);

    if not SectionExists(ProfileID) then begin
      LoadProfiles;
      exit;
    end;

    PROFILE APPLYING := true;
    TempRate := ReadInteger(ProfileID,'Compression_Rate_int',80);
    if (TempRate>0) and (TempRate<=100) then CompressionRate.Position :=
TempRate
    else CompressionRate.Position := 80;
    if ReadBool(ProfileID,'Compression_ReduceSize_bool',false) then begin
      RBtn_ChangeOGSize.Checked := true;
      RBtn_ModifiedSize.ItemIndex :=
ReadInteger(ProfileID,'Compression_NewSizeIndex_int',2);
    end else RBtn_KeepOGSize.Checked := true;
    case ReadInteger(ProfileID,'Destination_Index_int',0) of
      1: RBtn_CreateNewFolder.Checked := true;
      2: RBtn_CreateZIPFile.Checked := true;
    end;
  finally
    Free;
  end;
end;

```

```

{0:} else begin
    RBtn_OverwriteFiles.Checked := true;
    CBox_CreateBackup.Checked :=
ReadBool(ProfileID, 'Destination_MakeBackup_bool', false);
    end;
end;
if ReadBool(ProfileID, 'Metadata_DeleteMetadata_bool', false) then begin
    RBtn_RemoveAllMetadata.Checked := true;
    CBox_DeleteThumbnails.Checked :=
ReadBool(ProfileID, 'Metadata_DeleteThumbnails_bool', false);
    if ReadBool(ProfileID, 'Metadata_ChangeDate_bool', false) then begin
        CBox_ChangeDateData.Checked := true;
        FilesSelectedDate.Date :=
ReadDate(ProfileID, 'Metadata_NewDate_date', now);
        FilesSelectedTime.Time :=
ReadTime(ProfileID, 'Metadata_NewTime_time', now);
        end else CBox_ChangeDateData.Checked := false;
    end else RBtn_KeepAllMetadata.Checked := true;
    if ReadBool(ProfileID, 'Metadata_AddComment_bool', false) then begin
        CBox_AddGlobalComment.Checked := true;
        CommentText.Text := ReadString(ProfileID, 'Metadata_Comment_str', '');
    end else CBox_AddGlobalComment.Checked := false;
    if ReadBool(ProfileID, 'Metadata_AddCopyright_bool', false) then begin
        CBox_AddCopyright.Checked := true;
        CopyrightText.Text := ReadString(ProfileID, , '© Copyright ');
        case ReadInteger(ProfileID, 'Metadata_CopyrightPosition_int', 4) of
            // 1=TopLeft, 2=TopRight, 3=BottomLeft, 4=BottomRight
            1: CopyrightPositionMouseUp(CopyrightPositionTL, mbLeft, [], 1, 1);
            2: CopyrightPositionMouseUp(CopyrightPositionTR, mbLeft, [], 1, 1);
            3: CopyrightPositionMouseUp(CopyrightPositionBL, mbLeft, [], 1, 1);
            {4:} else CopyrightPositionMouseUp(CopyrightPositionBR, mbLeft, [], 1,
1);
        end;
        CopyrightFontDialog.Font.Name :=
ReadString(ProfileID, 'Metadata_CopyrightFontName_str', 'Tahoma');
        // FontName Validity Check :
        if (Screen.Fonts.IndexOf(CopyrightFontDialog.Font.Name) = -1) or
(CopyrightFontDialog.Font.Name='') then CopyrightFontDialog.Font.Name:='Tahoma';
        try CopyrightFontDialog.Font.Size :=
ReadInteger(ProfileID, 'Metadata_CopyrightFontSize_int', 10);
        except CopyrightFontDialog.Font.Size := 10 end;
        try CopyrightFontDialog.Font.Color :=
StringToColor(ReadString(ProfileID, 'Metadata_CopyrightFontColor_cl', 'clBlack'));
        except CopyrightFontDialog.Font.Color := clBlack end;
        CopyrightFontDialog.Font.Style := [];
        if ReadBool(ProfileID, 'Metadata_CopyrightFontStyleBold_bool', true) then
CopyrightFontDialog.Font.Style := CopyrightFontDialog.Font.Style + [fsBold];
        if ReadBool(ProfileID, 'Metadata_CopyrightFontStyleItalic_bool', false) then
CopyrightFontDialog.Font.Style := CopyrightFontDialog.Font.Style + [fsItalic];
        if ReadBool(ProfileID, 'Metadata_CopyrightFontStyleUnderline_bool', false)
then CopyrightFontDialog.Font.Style := CopyrightFontDialog.Font.Style +
[fsUnderline];
        if ReadBool(ProfileID, 'Metadata_CopyrightFontStyleStrikeOut_bool', false)
then CopyrightFontDialog.Font.Style := CopyrightFontDialog.Font.Style +
[fsStrikeOut];
        end else CBox_AddCopyright.Checked := false;
        finally
            Free;
            PROFILE_APPLYING := false;
        end;
    end;
end;
// ----- //
procedure TMainForm.SelectedProfileChange(Sender: TObject);
begin
    // ReadProfile (SelectedProfile.ItemIndex);
end;
//=====//
procedure TMainForm.ChangeControlsDisponibility(EnableControls: boolean);
begin

```

```

Panel_Files.Enabled := EnableControls;
Panel_Parameters.Enabled := EnableControls;
Panel_Destination.Enabled := EnableControls;
Panel_Compression.Enabled := EnableControls;
// SelectedProfile.Enabled := EnableControls;
Btn_BeginCompression.Enabled := EnableControls;

if EnableControls then Btn_CloseApp.Caption := 'Quitter'
else Btn_CloseApp.Caption := 'Annuler';

Application.ProcessMessages;
end;
// ----- //
procedure TMainForm.Btn_CloseAppClick(Sender: TObject);
begin
  if not IS_PROCESSING_IMAGES then Application.Terminate
  else ABORT_REQUESTED := true;
end;
// ----- //
procedure TMainForm.Btn_AboutAppClick(Sender: TObject);
begin
  AboutForm.ShowModal;
end;
//=====//
procedure TMainForm.Btn_BeginCompressionClick(Sender: TObject);
var
  i,j: integer;
  FileName, FinalDestinationFileName, FinalDestinationPath,
  OriginalFileExtension, ZIPFileName,
  BackupDestinationFileName, BackupDestinationPath, MetaDataSourceFileName:
string;
  xOG, yOG, xRED, yRED, tmp: integer;
  xratio, yratio, finalratio: extended;
  xText, yText: integer;
  FieldsCorrectlyCompleted, IsCorrectlyLoaded, IsCorrectlySaved,
IsLandscapeFormat: boolean;
  BMPTempImage3 : TBitmap;
  FormattedNewFileDateTime: string;
  ZIPPendingFiles: TStringList;
begin
  if FileList.Items.Count=0 then begin
    MessageBoxA(Handle, Pchar('Немає вибраних файлів'), Pchar('Помилка'),
MB_ICONSTOP + MB_SYSTEMMODAL + MB_SETFOREGROUND + MB_TOPMOST);
    exit;
  end;

  if ( (RBtn_OverwriteFiles.Checked) and (CBox_CreateBackup.Checked) and
(BackupPath.Text='') )
  or ( (RBtn_CreateNewFolder.Checked) and (CopyToDirectoryPath.Text='') )
  or ( (RBtn_CreateZIPFile.Checked) and (ExportZipFilePath.Text ='' ) ) then
begin
  MessageBoxA(Handle, Pchar('Вказаного для збереження файлів каталогу не
існує'), Pchar('Помилка'), MB_ICONSTOP + MB_SYSTEMMODAL + MB_SETFOREGROUND +
MB_TOPMOST);
  exit;
end;

Label_ProgressionStatus.Caption := 'Ініціалізація процесу стиснення...';
ChangeControlsDisponibility(false);

IS_PROCESSING_IMAGES := true;
ABORT_REQUESTED := false;

PBar_ProgressionStatus.Position := 0;
PBar_ProgressionStatus.Max := FileList.Count;
Application.ProcessMessages;

ErrorsList := TStringList.Create;
ErrorsList.Clear;

```

```

.....
if RBtn_OverwriteFiles.Checked then begin
  // "Перезаписуємі файли" вибір
  if CBox_CreateBackup.Checked then begin
    BackupDestinationPath := BackupPath.Text;
  end else begin // Навіть якби це не було запрошено споживачем, нам потрібно
робити копію файлу для того, щоб бути здатним перемістити метадані в новий файл
якщо необхідно
    BackupDestinationPath := TEMP_PATH;
  end;
end else begin
  if RBtn_CreateNewFolder.Checked then begin
    FinalDestinationPath :=
IncludeTrailingPathDelimiter(CopyToDirectoryPath.Text);
  end else if RBtn_CreateZIPFile.Checked then begin
    ZIPFileName := ExportZipFilePath.Text;
    ZIPPendingFiles := TStringList.Create;
    ZIPPendingFiles.Clear;
    FinalDestinationPath := TEMP_PATH + ExtractFileNameOnly(ZIPFileName) +
RandomString(10);
  end;
end;
FOR i := 0 TO FileList.Count-1 DO BEGIN
  if ABORT_REQUESTED then begin
    Label_ProgressionStatus.Caption := 'Скасування...';
    Application.ProcessMessages;
    for j := i to FileList.Count-1 do
ErrorsList.Add(FileList.Items[j]+' : Припинено користувачем. ');
      Break;
    end;
    FileName := FileList.Items[i];
    Label_ProgressionStatus.Caption := 'Обробка файлів
'+ExtractFileName(FileName);
    Application.ProcessMessages;
    if not FileExists(FileName) then begin
      ErrorsList.Add(FileName+' : файл недоступний. ');
      Continue; // Compression Process Abort : Next Picture
    end;

    IsCorrectlyLoaded := true;
    try
      if (UpperCase(ExtractFileExt(FileName))='.JPEG') or
(UpperCase(ExtractFileExt(FileName))='.ALGORITM_STYSKU') then begin
        JPEGImage1.LoadFromFile(FileName);
        Application.ProcessMessages;
        BMPTempImage1.Assign(JPEGImage1);
      end else if (UpperCase(ExtractFileExt(FileName))='.BMP') then
begin
        BMPTempImage1.LoadFromFile(FileName);
      end;
      if ((BMPTempImage1.Height<=1) or (BMPTempImage1.Width=0)) then
IsCorrectlyLoaded := false;
    except
      IsCorrectlyLoaded := false;
    end;
    if not IsCorrectlyLoaded then begin
      ErrorsList.Add(FileName+' : пошкоджений файл або некоректний
формат файлу');
      Continue;
    end;
    if RBtn_ChangeOGSize.Checked then begin
      xOG := BMPTempImage1.Width; yOG := BMPTempImage1.Height;

      if xOG>=yOG then IsLandscapeFormat:=true
      else begin //
        IsLandscapeFormat := false;
        tmp := xOG; xOG := yOG; yOG := tmp;
      end;
    end;
  end;
end;

```

```

xRED:=0; yRED:=0;
case RBtn_ModifiedSize.ItemIndex of
  0:begin xRED:=640; yRED:=480; end;
  1:begin xRED:=800; yRED:=600; end;
  2:begin xRED:=1024; yRED:=768; end;
  3:begin xRED:=1280; yRED:=1024; end;
  4:begin xRED:=1600; yRED:=1200; end;
  5:begin xRED:=2048; yRED:=1536; end;
end;
xratio:=(100*xRED)/xOG; yratio:=(100*yRED)/yOG;

if (xratio>=100) and (yratio>=100) then
BMPTempImage2.Assign(BMPTempImage1)
else begin
  finalratio := MinValue([xratio, yratio]);
  if not IsLandscapeFormat then begin
    tmp := xOG; xOG := yOG; yOG := tmp;
  end;
  xRED := round(xOG*finalratio/100); yRED :=
round(yOG*finalratio/100);

  if xRED=0 then xRED := 1; if yRED<=1 then yRED := 2;

  BMPTempImage3 := TBitmap.Create;
  try
    BMPTempImage3.Width := xRED;
    BMPTempImage3.Height := yRED;
    BMPTempImage3.PixelFormat := BMPTempImage1.PixelFormat;
    SetStretchBltMode(BMPTempImage3.Canvas.Handle, HALFTONE); //
Напівтон використаний для кращої вихідної якості образу
    StretchBlt(BMPTempImage3.Canvas.Handle, 0, 0, xRED, yRED,
BMPTempImage1.Canvas.Handle, 0, 0, xOG, yOG, SRCCOPY);
    BMPTempImage2.Width := xRED;
    BMPTempImage2.Height := yRED;
    BMPTempImage2.PixelFormat := BMPTempImage1.PixelFormat;
    BMPTempImage2.Assign(BMPTempImage3);
  finally
    BMPTempImage3.Free;
  end;
end;
end else begin
  BMPTempImage2.Assign(BMPTempImage1);
Reduction Required : Direct Assignment
end;
if CBox_AddCopyright.Checked then begin
  BMPTempImage2.Canvas.Brush.Style := BsClear;
  BMPTempImage2.Canvas.Font := CopyrightFontDialog.Font;
  xText:=30; yText:=30;
  if CopyrightPositionTL.Tag=1 then begin
    xText := 30;
    yText := 30;
  end else if CopyrightPositionBL.Tag=1 then begin
    xText := 30;
    yText := BMPTempImage2.Height-
(BMPTempImage2.Canvas.TextHeight(CopyrightText.Text)+30);
  end else if CopyrightPositionTR.Tag=1 then begin
    xText := BMPTempImage2.Width-
(BMPTempImage2.Canvas.TextWidth(CopyrightText.Text)+30);
    yText := 30;
  end else if CopyrightPositionBR.Tag=1 then begin
    xText := BMPTempImage2.Width-
(BMPTempImage2.Canvas.TextWidth(CopyrightText.Text)+30);
    yText := BMPTempImage2.Height-
(BMPTempImage2.Canvas.TextHeight(CopyrightText.Text)+30);
  end;
  BMPTempImage2.Canvas.TextOut(xText, yText, CopyrightText.Text);
end;
if RBtn_OverwriteFiles.Checked then begin

```

```

        OriginalFileExtension := ExtractFileExt(FileName);
        ForceDirectories(BackupDestinationPath);
        BackupDestinationFileName :=
FindAvailableFileName(BackupDestinationPath, ExtractFileNameOnly(FileName),
OriginalFileExtension);
        if not
CopyFile(PChar(FileName), PChar(BackupDestinationFileName), true) then begin
        ErrorsList.Add(FileName+' : Пошкоджений файл, або некоректний
формат файлу');
        Continue;
        end;
        if (OriginalFileExtension='.jpeg') then
RenameFile(FileName, ChangeFileExt(FileName, '.Algoritm_stysku'));
        Application.ProcessMessages;
        FinalDestinationFileName := ChangeFileExt(FileName, '.shm');
        end else begin

        ForceDirectories(FinalDestinationPath);
        Application.ProcessMessages;
        FinalDestinationFileName :=
FindAvailableFileName(FinalDestinationPath, ExtractFileNameOnly(FileName),
'.shm');

        end;
        IsCorrectlySaved := true;
        JPEGImage2 := TJpegImage.Create;
        try
        try
            JPEGImage2.Assign(BMPTempImage2);
            Application.ProcessMessages;
            JPEGImage2.CompressionQuality := CompressionRate.Position;
            JPEGImage2.Compress;
            JPEGImage2.SaveToFile(FinalDestinationFileName);
        except
            IsCorrectlySaved:= false;
        end;
        finally
            JPEGImage2.Free;
        end;
        if not IsCorrectlySaved then begin
            ErrorsList.Add(FileName+' : Файл не може бути стиснутий. Помилка
збереження. ');
            Continue;
        end;

        // if (RBtn_OverwriteFiles.Checked) and
(OriginalFileExtension='.bmp') then DeleteFile(FileName);
        if RBtn_CreateZIPFile.Checked then
ZIPPendingFiles.Add(FinalDestinationFileName);
        Application.ProcessMessages;

        if RBtn_OverwriteFiles.Checked then
MetadataSourceFileName:=BackupDestinationFileName
        else MetadataSourceFileName := FileName;
        if RBtn_KeepAllMetadata.Checked then begin
            LaunchAndWait(''+EXIFTOOL_PATH+' -TagsFromFile
''+MetadataSourceFileName+' -all:all ''+FinalDestinationFileName+' -P -q -m -
overwrite_original', SW_HIDE);
            Application.ProcessMessages;
        end;
        if CBox_AddGlobalComment.Checked then begin
            LaunchAndWait(''+EXIFTOOL_PATH+' -
UserComment='''+CommentText.Text+''' -XPCOMMENT='''+CommentText.Text+''' -
xmp:UserComment='''+CommentText.Text+''' -Comment='''+CommentText.Text+'''
''+FinalDestinationFileName+' -P -q -m -L -overwrite_original', SW_HIDE);
            Application.ProcessMessages;
        end;
        if CBox_AddCopyright.Checked then begin

```

```

        LaunchAndWait(''+EXIFTOOL_PATH+''' -
Copyright=''+CopyrightText.Text+''' '''+FinalDestinationFileName+''' -P -q -m -L -
overwrite_original',SW_HIDE);
        Application.ProcessMessages;
    end;
    if RBtn_RemoveAllMetadata.Checked then begin
        if not CBox_DeleteThumbnails.Checked then begin
            //
            LaunchAndWait(''+EXIFTOOL_PATH+''' -TagsFromFile
'''+MetaDataSourceFileName+''' -ThumbnailImage '''+FinalDestinationFileName+''' -P -
q -m -overwrite_original',SW_HIDE);
            Application.ProcessMessages;
        end;
        if (RBtn_RemoveAllMetadata.Checked) and
(CBox_ChangeDateData.Checked) then begin
            FormattedNewFileDateTime := FormatDateTime('yyyy:mm:dd
hh:mm:ss', FilesSelectedDate.DateTime);
            LaunchAndWait(''+EXIFTOOL_PATH+''' -
DateTimeOriginal=''+FormattedNewFileDateTime+''' '
CreateDate=''+FormattedNewFileDateTime+'Z" '
MetadataDate=''+FormattedNewFileDateTime+'Z" '
ModifyDate=''+FormattedNewFileDateTime+''' '
'''+FinalDestinationFileName+''' -P -q -m -overwrite_original',SW_HIDE);
            Application.ProcessMessages;
            // File DateTime Corruption (CreationDate, LastModifiedDate,
LastAccessedDate)
            CorruptFileDateTime(FinalDestinationFileName,
FilesSelectedDate.DateTime);
        end;
    end;

    if (RBtn_OverwriteFiles.Checked) and (not CBox_CreateBackup.Checked)
then DeleteFile(BackupDestinationFileName);
    PBar_ProgressionStatus.Position := i+1;
END;

    if (RBtn_CreateZIPFile.Checked) and (not ABORT_REQUESTED) then begin

        if FileExists(ZIPFileName) and (not DeleteFile(ZIPFileName)) then begin
            ErrorsList.Add('Фатальна помилка : не вдалося створити ZIP архів
(''+ExtractFileName(ZIPFileName)+'')');
            ErrorsList.Add('Переконайтеся в тому, що шлях збереження фалів задано
вірно');
            ABORT_REQUESTED := false;
        end else begin

            ZipComponent := TZip.create(self);
            try
                try
                    ZipComponent.FileName := ZIPFileName;
                    if CBox_AddGlobalComment.Checked then ZipComponent.ZipComment :=
CommentText.Text
                    else ZipComponent.ZipComment := '';
                    ZipComponent.FileSpecList := ZipPendingFiles;
                    ZipComponent.AddOptions := [aoUpdate];
                    if (ZipComponent.Add <> ZipPendingFiles.Count) then begin
                        if ZipComponent.Cancelled then begin
                            ABORT_REQUESTED := true;
                            ErrorsList.Add(ZIPFileName+' : помилка створення ZIP архіву');
                        end else begin
                            ABORT_REQUESTED := false;
                            ErrorsList.Add('деякі файли не вдалося додати в ZIP
архів'''+ExtractFileName(ZIPFileName)+'''');
                        end;
                    end;
                end;
            end;
        end;
    end;
end;

```

```

    except
        ErrorsList.Add(('+ExtractFileName(ZIPFileName)+'));
        ErrorsList.Add();
        ABORT_REQUESTED := false;
    end;
finally
    ZipComponent.Free;
end;
if CBox_ChangeDateData.Checked then CorruptFileDateTime(ZIPFileName,
FilesSelectedDate.DateTime);
end;
end;
PBar_ProgressionStatus.Position := PBar_ProgressionStatus.Max;
Label_ProgressionStatus.Caption := 'Відміна процесу стиснення користувачем';
Application.ProcessMessages;
ErrorForm.Caption := 'Користувач відмінив процес стиснення';
if RBtn_OverwriteFiles.Checked then ErrorForm.ErrorTitleMsg.Caption :=
'Деякі файли не були стиснуті '+ #13 +' вже оброблені файли не будуть
відновлені.'
else ErrorForm.ErrorTitleMsg.Caption := 'Деякі файли не були стислі '+ #13
+' вже оброблені файли не будуть видалені';
ErrorForm.FilesConcernedMemo.Lines := ErrorsList;
ErrorForm.ShowModal;
IS_PROCESSING_IMAGES := false;
ABORT_REQUESTED := false;
ChangeControlsDisponibility(true);
if RBtn_CreateZIPFile.Checked then ZipPendingFiles.Free;
ErrorsList.Free;
Application.ProcessMessages;
DeleteDirectory(TEMP_PATH);
Application.ProcessMessages;
exit;
end else begin
    Btn_CloseApp.Caption := 'Скасування'#10#13'неможливе';
    Btn_CloseApp.Enabled := false;
end;
PBar_ProgressionStatus.Position := PBar_ProgressionStatus.Max;
if (ErrorsList.Text<>'') then begin
    Label_ProgressionStatus.Caption := 'Стиснення відбулося з помилками';
    Application.ProcessMessages;
    ErrorForm.Caption := 'Помилка при компресії файлів';
    ErrorForm.ErrorTitleMsg.Caption := 'Деякі файли не можуть бути стиснуті
правильно '+ #13 +' Перевірте формат і доступність:';
    ErrorForm.FilesConcernedMemo.Lines := ErrorsList;
    ErrorForm.ShowModal;
end else begin
    Label_ProgressionStatus.Caption := 'Стиснення успішно завершилося';
    Application.ProcessMessages;
    MessageBoxA(MainForm.Handle, Pchar('Всі файли було успішно
стиснуто'), Pchar('Стиснення заверрене'), MB_ICONINFORMATION + MB_SYSTEMMODAL +
MB_SETFOREGROUND + MB_TOPMOST);
end;
IS_PROCESSING_IMAGES := false;
Btn_CloseApp.Enabled := true;
ChangeControlsDisponibility(true);
Btn_CloseApp.Enabled := true;
if RBtn_CreateZIPFile.Checked then ZipPendingFiles.Free;
ErrorsList.Free;
Application.ProcessMessages;
DeleteDirectory(TEMP_PATH);
Application.ProcessMessages;
end;
procedure TMainForm.FormCreate(Sender: TObject);
var
    RES: TResourceStream;
begin
    DragAcceptFiles(FileList.Handle, true); // Допускаються Drag'n Drop процедури
    FilesSelectedDate.Date := now;
    FilesSelectedTime.Time := now;

```

```

APP_PATH :=
IncludeTrailingPathDelimiter(ExtractFilePath(Application.ExeName));
EXIFTOOL_PATH := APP_PATH+'exiftool.exe';
TEMP_PATH :=
IncludeTrailingPathDelimiter(ExtractFilePath(Application.ExeName)) + '$temp$\';
PROFILES_CFGFILE := APP_PATH+'CompressionProfiles.jcfg';

if not FileExists(EXIFTOOL_PATH) then begin
  try
    RES := TResourceStream.Create(0, 'EXIFTOOL', 'EXEFILE');
    try
      RES.SaveToFile(EXIFTOOL_PATH);
      Application.ProcessMessages;
    finally
      RES.Free;
    end;
  except
    MessageBox(0, Pchar('Виникла помилка при ініціалізації програми, яка
перешкоджає її виконанню. '+ #13 +' Файл "exiftool.exe" не було
знайдено. '), Pchar('Помилка'), MB_ICONSTOP + + MB_ICONINFORMATION +
MB_SYSTEMMODAL + MB_SETFOREGROUND + MB_TOPMOST);
    Application.Terminate;
  end;
end;
if not FileExists(APP_PATH+'ZipDll.dll') then begin
  try
    RES := TResourceStream.Create(0, 'ZIPDLL', RT_RCDATA);
    try
      RES.SaveToFile(APP_PATH+'ZipDll.dll');
      Application.ProcessMessages;
    finally
      RES.Free;
    end;
  except
    MessageBox(0, Pchar('Виникла помилка при ініціалізації програми, яка
перешкоджає її виконанню. '+ #13 +' Файл "ZipDll.dll" не було
знайдено. '), Pchar('Помилка'), MB_ICONSTOP + + MB_ICONINFORMATION + MB_SYSTEMMODAL
+ MB_SETFOREGROUND + MB_TOPMOST);
    Application.Terminate;
  end;
end;

LoadProfiles();

end;
// ----- //
procedure TMainForm.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
begin
  if IS_PROCESSING_IMAGES then CanClose:=false;
end;
// ----- //
procedure TMainForm.Timer1Timer(Sender: TObject);
begin
  //DisplayForm.CompressionRate
end;
initialization
  Randomize;
  JPEGImage1 := TJpegImage.Create;
  BMPTempImage1 := TBitmap.Create;
  BMPTempImage2 := TBitmap.Create;
// ----- //
finalization
  BMPTempImage2.Free;
  BMPTempImage1.Free;
  JPEGImage1.Free;
  Application.ProcessMessages;
  DeleteDirectory(TEMP_PATH);
end.

```

Preview.pas - файл передперегляду зображень

```

unit Preview;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ComCtrls, ExtCtrls, Jpeg, Buttons;

type
  TDisplayForm = class(TForm)
    Panel_DisplayImage: TPanel;
    DisplayImage: TImage;
    Panel_DisplaySettings: TGroupBox;
    RBtn_ShowOriginalImage: TRadioButton;
    RBtn_ShowPreviewImage: TRadioButton;
    CompressionBar: TTrackBar;
    Label_CompressionRate: TLabel;
    Label_Title1: TLabel;
    ZoomFactorIndex: TTrackBar;
    Label_Title2: TLabel;
    Label_ZoomFactor: TLabel;
    RefreshAdvice_Label: TLabel;
    RefreshAdvice_Fond: TShape;
    RefreshAdvice_Image: TImage;
    ProcessingFile_Image: TImage;
    Label_OriginalFileSize: TLabel;
    Label_CompressedFileSize: TLabel;
    Button1: TButton;
    procedure FormCreate(Sender: TObject);
    procedure ImageMouseDown(Sender: TObject; Button: TMouseButton; Shift:
TShiftState; X, Y: Integer);
    procedure ImageMouseMove(Sender: TObject; Shift: TShiftState; X, Y:
Integer);
    procedure CompressionRateChange(Sender: TObject);
    procedure RefreshDisplay(Sender: TObject);
    procedure ZoomFactorIndexChange(Sender: TObject);
    procedure LoadPreview(FileName: string);
    procedure Button1Click(Sender: TObject);
  private
    { Private Declarations }
    function GetZoomFactorFromIndex(Index: integer): integer;
  public
    { Public Declarations }
  end;

var
  DisplayForm: TDisplayForm;
  CursorStartPosX, CursorStartPosY : integer;

  CURRENTLY_DISPLAYED_FILE_NAME : string; // \
  CURRENTLY_USED_COMPRESSION_RATE : integer; // >- Збережені
частини інформації
  CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE : string; // /

implementation

uses Main;

{$R *.dfm}
procedure TDisplayForm.FormCreate(Sender: TObject);
begin
  // DoubleBuffered := true;
  // Panel_DisplayImage.DoubleBuffered := true;
end;
function TDisplayForm.GetZoomFactorFromIndex(Index: integer): integer;
begin

```

```

case Index of
  1: result:=12;
  2: result:=25;
  3: result:=50;
  5: result:=200;
  6: result:=400;
  7: result:=800;
else result:=100;
end;
end;
// ----- //
procedure TDisplayForm.ImageMouseDown(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
  CursorStartPosY := Y;
  CursorStartPosX := X;
end;
// ----- //
procedure TDisplayForm.ImageMouseMove(Sender: TObject; Shift: TShiftState; X, Y:
Integer);
begin
  if (ssLeft in Shift) then begin
    DisplayImage.Top := DisplayImage.Top - (CursorStartPosY-Y);
    DisplayImage.Left := DisplayImage.Left - (CursorStartPosX-X);
    if DisplayImage.Top < Panel_DisplayImage.Height - DisplayImage.Height then
DisplayImage.Top := Panel_DisplayImage.Height - DisplayImage.Height;
    if DisplayImage.Left < Panel_DisplayImage.Width - DisplayImage.Width then
DisplayImage.Left := Panel_DisplayImage.Width - DisplayImage.Width;
    if DisplayImage.Top > 0 then DisplayImage.Top := 0;
    if DisplayImage.Left > 0 then DisplayImage.Left := 0;
  end;
end;
// ----- //
procedure TDisplayForm.ZoomFactorIndexChange(Sender: TObject);
var
  ZoomFactor: integer;
begin
  ZoomFactor := GetZoomFactorFromIndex(ZoomFactorIndex.Position);
  Label_ZoomFactor.Caption := IntToStr(ZoomFactor)+' %';
  DisplayImage.Height := round((ZoomFactor*DisplayImage.Picture.Height)/100);
  DisplayImage.Width := round((ZoomFactor*DisplayImage.Picture.Width)/100);
  if (DisplayImage.Height + DisplayImage.Top) < Panel_DisplayImage.Height then
DisplayImage.Top := (Panel_DisplayImage.Height - DisplayImage.Height);
  if (DisplayImage.Width + DisplayImage.Left) < Panel_DisplayImage.Width then
DisplayImage.Left := (Panel_DisplayImage.Width - DisplayImage.Width);
  if DisplayImage.Top > 0 then DisplayImage.Top := 0;
  if DisplayImage.Left > 0 then DisplayImage.Left := 0;
end;
// ----- //
procedure TDisplayForm.CompressionRateChange(Sender: TObject);
var
  AdviceControlsAvailability: boolean;

begin
  Label_CompressionRate.Caption := IntToStr(CompressionRate.Position) + ' %';

  AdviceControlsAvailability :=
(CURRENTLY_USED_COMPRESSION_RATE<>CompressionRate.Position) and
RBtn_ShowPreviewImage.Checked;
  RefreshAdvice_Image.Visible := AdviceControlsAvailability;
  RefreshAdvice_Label.Visible := AdviceControlsAvailability;
  RefreshAdvice_Fond.Visible := AdviceControlsAvailability;

  if not AdviceControlsAvailability then
    Label_CompressedFileSize.Caption :=
CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE
  else Label_CompressedFileSize.Caption := '( ??? Kb )';

```

```

MainForm.CompressionRate.Position:=DisplayForm.CompressionRate.Position;
end;
procedure TDisplayForm.LoadPreview(FileName: string);
begin
    CURRENTLY_DISPLAYED_FILE_NAME := FileName;
    CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE := '( ??? Kb )';
    Label_CompressedFileSize.Caption := CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE;
    RefreshDisplay(MainForm);
end;
procedure TDisplayForm.RefreshDisplay(Sender: TObject);
var
    LoadingFailure : boolean;
    JPEGImg1 : TJpegImage;
    JPEGImg2 : TJpegImage;
    BMPTempImg : TBitmap;
    TempStream : TMemoryStream;
    ImgFile : file of Byte;
begin
    LoadingFailure := false;
    DisplayForm.Cursor := crHourGlass;

    RefreshAdvice_Image.Visible := false;
    RefreshAdvice_Label.Visible := false;
    RefreshAdvice_Fond.Visible := false;
    ProcessingFile_Image.Visible := true;
    Application.ProcessMessages;

    CURRENTLY_USED_COMPRESSION_RATE := CompressionRate.Position;

    if RBtn_ShowOriginalImage.Checked then begin
CompressionRate.Enabled:=False;
        try
            DisplayImage.Picture.LoadFromFile(CURRENTLY_DISPLAYED_FILE_NAME);
        except
            MessageBoxA(MainForm.Handle, Pchar('Не вдається прочитати зображення.
Помилка формату. '), Pchar('Помилка'), MB_ICONSTOP + MB_SYSTEMMODAL +
MB_SETFOREGROUND + MB_TOPMOST);
            LoadingFailure := true;
        end;
        if LoadingFailure then begin
            if not (Sender = MainForm) then Close;
            DisplayForm.Cursor := crDefault;
            exit;
        end;
    end else begin
CompressionRate.Enabled:=True;
        JPEGImg1 := TJpegImage.Create;
        BMPTempImg := TBitmap.Create;
        JPEGImg2 := TJpegImage.Create;
        TempStream := TMemoryStream.Create;
        try
            if (UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.JPEG') or
(UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.ALGORITM_STYSKU') or
(UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.SHM') then begin
//-----
                JPEGImg1.LoadFromFile(CURRENTLY_DISPLAYED_FILE_NAME); // Завантаження
                файлу
                Application.ProcessMessages;
                BMPTempImg.Assign(JPEGImg1);
            end else if
(UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.BMP') then begin
                BMPTempImg.LoadFromFile(CURRENTLY_DISPLAYED_FILE_NAME); //
                Завантаження файлу
            end;
            JPEGImg2.Assign(BMPTempImg);
            Application.ProcessMessages;
        end;
    end;
end;

```

```

        JPEGImg2.CompressionQuality:= CompressionRate.Position;
        JPEGImg2.Compress;
        JPEGImg2.SaveToStream(TempStream);
        TempStream.Position:=0;
        JPEGImg2.LoadFromStream(TempStream);
        CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE := '(
'+Inttostr(TempStream.Size div 1024)+' Kb )';
        Label_CompressedFileSize.Caption :=
CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE;
        DisplayImage.Picture.Assign(JPEGImg2);
        finally
            TempStream.Free;
            JPEGImg2.Free;
            BMPTempImg.Free;
            JPEGImg1.Free;
        end;
    except
        MessageBoxA(MainForm.Handle,Pchar('Не вдається прочитати зображення.
Помилка формату. '),Pchar('Помилка'),MB_ICONSTOP + MB_SYSTEMMODAL +
MB_SETFOREGROUND + MB_TOPMOST);
        LoadingFailure := true;
    end;
    if LoadingFailure then begin
        if not (Sender = MainForm) then Close;
        DisplayForm.Cursor := crDefault;
        exit;
    end;

end;
DisplayForm.Cursor := crDefault;
ProcessingFile_Image.Visible := false;
if (Sender = MainForm) then begin
    AssignFile(ImgFile, CURRENTLY_DISPLAYED_FILE_NAME);
    Reset(ImgFile);
    Label_OriginalFileSize.Caption := '( '+Inttostr(FileSize(ImgFile) div
1024)+' Kb )';
    CloseFile(ImgFile);
    DisplayImage.Width := DisplayImage.Picture.Width;
    DisplayImage.Height := DisplayImage.Picture.Height;
    DisplayImage.Top := 0;
    DisplayImage.Left := 0;
    ZoomFactorIndex.Position := 4; // Первісний розмір зображення
    ShowModal;
end;

end;
procedure TDisplayForm.Button1Click(Sender: TObject);
var
    LoadingFailure : boolean;
    JPEGImg1 : TJpegImage;
    JPEGImg2 : TJpegImage;
    BMPTempImg : TBitmap;
    TempStream : TMemoryStream;
    ImgFile : file of Byte;
begin
    LoadingFailure := false;
    DisplayForm.Cursor := crHourGlass;

    RefreshAdvice_Image.Visible := false;
    RefreshAdvice_Label.Visible := false;
    RefreshAdvice_Fond.Visible := false;
    ProcessingFile_Image.Visible := true;
    Application.ProcessMessages;

    CURRENTLY_USED_COMPRESSION_RATE := CompressionRate.Position;
    if RBtn_ShowOriginalImage.Checked then begin
        try
            DisplayImage.Picture.LoadFromFile(CURRENTLY_DISPLAYED_FILE_NAME);
        except

```

```

        MessageBoxA(MainForm.Handle, Pchar('Не вдається прочитати зображення.
Помилка формату. '), Pchar('Помилка'), MB_ICONSTOP + MB_SYSTEMMODAL +
MB_SETFOREGROUND + MB_TOPMOST);
        LoadingFailure := true;
    end;
    if LoadingFailure then begin
        if not (Sender = MainForm) then Close;
        DisplayForm.Cursor := crDefault;
        exit;
    end;

end else begin
    JPEGImg1 := TJpegImage.Create;
    BMPTempImg := TBitmap.Create;
    JPEGImg2 := TJpegImage.Create;
    TempStream := TMemoryStream.Create;
    try
        if (UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.JPEG') or
(UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.ALGORITM_STYSKU') or
(UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.SHM') then begin
//-----
            JPEGImg1.LoadFromFile(CURRENTLY_DISPLAYED_FILE_NAME); // Завантаження
файлу
            Application.ProcessMessages;
            BMPTempImg.Assign(JPEGImg1);
        end else if
(UpperCase(ExtractFileExt(CURRENTLY_DISPLAYED_FILE_NAME))='.BMP') then begin
            BMPTempImg.LoadFromFile(CURRENTLY_DISPLAYED_FILE_NAME); //
Завантаження файлу
            end;
            JPEGImg2.Assign(BMPTempImg);
            Application.ProcessMessages;
            JPEGImg2.CompressionQuality:= CompressionRate.Position;
            JPEGImg2.Compress;
            JPEGImg2.SaveToStream(TempStream);
            TempStream.Position:=0;
            JPEGImg2.LoadFromStream(TempStream);
            CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE := '(
'+Inttostr(TempStream.Size div 1024)+' Kb)';
            Label_CompressedFileSize.Caption :=
CURRENTLY_DETERMINED_COMPRESSED_FILE_SIZE;
            DisplayImage.Picture.Assign(JPEGImg2);
        finally
            TempStream.Free;
            JPEGImg2.Free;
            BMPTempImg.Free;
            JPEGImg1.Free;
        end;
    except
        MessageBoxA(MainForm.Handle, Pchar('Не вдається прочитати зображення.
Помилка формату.'+#13+'Le fichier peut avoir йтй effacy ou ktre
corrompu. '), Pchar('Помилка'), MB_ICONSTOP + MB_SYSTEMMODAL + MB_SETFOREGROUND +
MB_TOPMOST);
        LoadingFailure := true;
    end;
    if LoadingFailure then begin
        if not (Sender = MainForm) then Close;
        DisplayForm.Cursor := crDefault;
        exit;
    end;

end;

DisplayForm.Cursor := crDefault;
ProcessingFile_Image.Visible := false;

if (Sender = MainForm) then begin
    AssignFile(ImgFile, CURRENTLY_DISPLAYED_FILE_NAME);
    Reset(ImgFile);

```

```
Label_OriginalFileSize.Caption := '( '+Inttostr(FileSize(ImgFile) div
1024)+' Kb )';
CloseFile(ImgFile);
DisplayImage.Width := DisplayImage.Picture.Width;
DisplayImage.Height := DisplayImage.Picture.Height;
DisplayImage.Top := 0;
DisplayImage.Left := 0;
ZoomFactorIndex.Position := 4; // Первісний розмір зображення
ShowModal;
end;

end;

end.
```

КБПЗ - 2023

Error.pas - файл обработки ошибок

```
unit Error;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, Buttons;

type
  TErrorForm = class(TForm)
    ErrorTitleMsg: TLabel;
    FilesConcernedMemo: TMemo;
    CloseBtn: TBitBtn;
    ClipboardCopyBtn: TBitBtn;
    procedure ClipboardCopyBtnClick(Sender: TObject);
    procedure CloseBtnClick(Sender: TObject);
  private
    { Private Declarations }
  public
    { Public Declarations }
  end;

var
  ErrorForm: TErrorForm;

implementation

{$R *.dfm}

procedure TErrorForm.CloseBtnClick(Sender: TObject);
begin
  Close;
end.
```

about.pas - файл довідки

```
unit about;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, jpeg, ExtCtrls;

type
  TForm5 = class(TForm)
    Memo1: TMemo;
    Button1: TButton;
    Image1: TImage;
    procedure FormCreate(Sender: TObject);
    procedure Button1Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  Form5: TForm5;

implementation

{$R *.dfm}

procedure TForm5.FormCreate(Sender: TObject);
begin
  Memo1.Clear;
  Memo1.Lines.Add(' МАГІСТЕРСЬКА РОБОТА');
  Memo1.Lines.Add('');
  Memo1.Lines.Add(' на тему:');
  Memo1.Lines.Add('');
  Memo1.Lines.Add(' Програмне забезпечення центру стискання та перетворення  
форматів зображень ');
  Memo1.Lines.Add('');
  Memo1.Lines.Add(' Керівник: Смірнов О.А. ');
  Memo1.Lines.Add('');
  Memo1.Lines.Add(' Розробив: студент Бабаєв Роман Ілгарович');
  Memo1.Lines.Add(' гр. КН-22М-1');
  Memo1.Lines.Add('');
  Memo1.Lines.Add(' Кропивницький 2023');
  Memo1.Lines.Add('');
end;

procedure TForm5.Button1Click(Sender: TObject);
begin
  Form5.Close;
end;
end.
```