

## ЗМІСТ

ВСТУП .....	3
1. АНАЛІЗ ОСНОВНИХ ЗАСОБІВ ПРОГРАМУВАННЯ МОВОЮ C++ .....	4
1.1 Історія та функції ООП.....	4
1.2. C++.....	6
1.3 Типові рішення ООП для баз даних.....	11
2. РОЗРОБКА ПРОГРАМИ ВИКОНАННЯ ЗАВДАННЯ.....	14
2.1 Аналіз предметної області “Книжковий магазин” .....	14
2.2 Постановка завдання .....	16
2.3 Вхідні дані.....	16
2.4 Розробка моделі бази даних .....	16
2.5 Розробка алгоритму вирішення задач, що виникли під час виконання роботи.....	23
2.5.1 Додавання запису в базу.....	23
2.5.2 Видалення запису з бази.....	24
2.5.3 Редагування запису у базі .....	24
2.5.4 Формування списку на основі первинних документів .....	25
2.5.5 Запити на виведення даних, що відповідають певним умовам.....	25
2.5.6 Форматований вивід списку .....	26
2.5.7 Відбір та групування списку унікальних даних на основі загального списку даних.....	27
3. КОНТРОЛЬНИЙ ПРИКЛАД.....	29
Ситуація 1: додавання надходження.....	29
Ситуація 2: реалізація книг .....	30
Ситуація 3: ревізія магазину і бази даних .....	30
Ситуація 4: приймання на роботу нового працівника.....	31
Ситуація 5: звільнення працівника.....	31
Ситуація 6: зміна статусу існуючого продавця.....	31
Ситуація 7: оформлення для покупця дисконтної програми.....	32

					<b>КР.ІП-25.00.00.000 ПЗ</b>			
Змн.	Лист	№ докум.	Підпис	Дата				
Розроб.		Корнута Ю. В.			Розробка програмного рішення для консольної бази даних по предметній області «Книжковий магазин» засобами C++	Літ.	Арк.	Акрushів
Перевір.		Шекета В. І.					1	103
Реценз.						ІФНТУНГ, ІП-19-2		
Н. Контр.								
Затверд.								

ВИСНОВКИ .....	33
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	34
Додаток А Код програми мовою C++ .....	36
class_books.h .....	36
class_people.h .....	43
class_purchase.h.....	60
class_reports.h.....	72
MENU.cpp .....	81
Додаток Б Результати контрольного прикладу.....	88
Додаток В Діаграми класів .....	101

ВСТУП

					КР.ІП-25.00.00.000 ПЗ	Арк.
						3
Зм.	Арк.	№ докум.	Підп.	Дата		

# 1. АНАЛІЗ ОСНОВНИХ ЗАСОБІВ ПРОГРАМУВАННЯ МОВОЮ C++

## 1.1. Історія та функції ООП

Об'єктно-орієнтоване програмування (ООП) – це модель програмування яка базується на ствердженні того, що програма це сукупність об'єктів які взаємодіють між собою. Кожен об'єкт в цій моделі є незалежним, і він здатний отримувати, обробляти дані та відправляти ці дані іншим об'єктам. В ООП використано моделі успадкування, модульності, поліморфізму та інкапсуляції.

Основним поняттям ООП є об'єкт. Об'єкт можна визначити як певну сукупність даних та методів роботи з ними. Для класифікації об'єктів у ООП використовують класи. Клас служить зразком для створення об'єкту, тобто об'єкт є нічим іншим, ніж копією класу.

Кожен об'єкт має процедури і функції, які служать для роботи з даними об'єкта. Ці процедури і функції називаються методами.

Існування ООП можливе завдяки трьом основним парадигмам на яких базується саме ООП:

- Інкапсуляція. Також відома як приховування даних. Зміст інкапсуляції полягає у приховуванні від зовнішнього користувача деталей реалізації об'єкта, замість цього надаючи інтерфейс взаємодії з ним.

- Успадкування. Це означає, що об'єкти (класи) можуть переймати деякі властивості у своїх прабатьків. Зазвичай картина така: це призводить до повторного використання вже написаного одного разу коду. Підкласи успадковують атрибути та поведінку своїх батьківських класів, і можуть мати нові власні атрибути. Тобто утворюється ієрархія з класів, де від основного класу походять усі інші класи.

- Поліморфізм означає залежність поведінки від класу, в якому ця поведінка викликається, тобто, два або більше класів можуть реагувати по різному на однакові повідомлення. Це спричинене зміною в одного з класів якогось методу, шляхом запису іншого алгоритму.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						4
Зм.	Арк.	№ докум.	Підп.	Дата		

Об'єктно-орієнтований підхід до програмування є пріоритетним при створюванні переважної більшості програмних проектів.

В остаточному вигляді кожна програма є набором інструкцій для процесора. І чим вище є рівень мови, тим у більш простій формі записуються одні й ті ж самі дії. З нарощуванням обсягу програм стає потрібним структурувати інформацію, виокремлювати в ній головне та відкидати несуттєве. Цей процес називається підвищенням ступеня абстракції програми.

Першим кроком до підвищення абстракції є використання функцій, що дозволяє після написання та налагодження функції дистанціюватись від деталей її реалізації, оскільки для виклику функції треба знати лише її інтерфейс. Наступний крок – оголошення власних типів даних, які дозволяють структурувати та групувати інформацію, подаючи її в більш природному вигляді. Оскільки для роботи з власними типами даних потрібні спеціальні функції, тому вважається за природне згрупувати їх разом з оголошенням цих типів в одному місці програми і у певний спосіб відокремити від решти її частин. Отже, об'єднання в модулі оголошень типів даних та функцій, призначених для роботи з цими типами, разом із приховуванням від користувача модуля несуттєвих деталей, є подальшим розвитком структуризації програми.

Введення поняття класу є розвитком ідей модульності. У класі поєднуються структури даних і функції їхнього опрацювання. Клас використовується лише через його інтерфейс – деталі реалізації для користувача класу є несуттєві.

Ідея класів відображає будову об'єктів реального світу. Адже ж кожен об'єкт чи процес має набір певних характеристик чи відмінностей, інакше кажучи, певні властивості й поведінку. А користуватись об'єктами можна, не знаючи їхнього внутрішнього зображення. Наприклад, керувати автомобілем можна, не маючи уявлення про будову його двигуна чи будь-яких інших його частин.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						5
Зм.	Арк.	№ докум.	Підп.	Дата		

## 1.2. C++

C++ - мова програмування високого рівня з підтримкою кількох парадигм програмування: об'єктно-орієнтованої, узагальненої та процедурної. Розроблена Б'ярном Страуструпом в AT&T Bell Laboratories 1979 року та початково отримала назву «Сі з класами». Згодом Страуструп перейменував мову на C++ у 1983 р. Базується на мові С.

У 1990-х роках C++ стала однією з найуживаніших мов програмування загального призначення. Мову використовують для системного програмування, розробки програмного забезпечення, написання драйверів, потужних серверних та клієнтських програм, а також для розробки розважальних програм, наприклад, відеоігор. C++ суттєво вплинула на інші популярні сьогодні мови програмування: C# та Java.

Мова Сі++ багато в чому є надмножиною Сі. Нові можливості Сі++ включають оголошення у вигляді виразів, перетворення типів у вигляді функцій, оператори `new` і `delete`, тип `bool`, посилання, розширене поняття константності та змінності, функції, що підставляються, аргументи за умовчанням, перевизначення, простори імен, класи, перевизначення операторів, шаблони, оператор `::`, обробку винятків, динамічну ідентифікацію і багато що інше. Сі++ є також мовою строгого типування і накладає більше вимог щодо дотримання типів, порівняно з Сі.

У Сі++ з'явилися коментарі у вигляді подвійної косої риски, які були в попереднику Сі — мові BCPL.

Деякі особливості Сі++ пізніше були перенесені в Сі, наприклад ключові слова `const` і `inline`, оголошення в циклах `for` і коментарі в стилі Сі++. У пізніших реалізаціях Сі також були представлені можливості, яких немає в Сі++, наприклад макроси `vararg` і покращена робота з масивами-параметрами.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						6
Зм.	Арк.	№ докум.	Підп.	Дата		

В мові С основним способом організації даних були структури. Структура складається з набору полів, які ніяк не захищені. Якщо елементи структури мають змінну довжину, їх представляють у вигляді вказівників. Виділення і звільнення пам'яті під ці вказівники робляться вручну.

Така реалізація небезпечна і неефективна з багатьох причин, але мова Сі++, використовуючи ООП, усуває всі ці проблеми.

З публічними полями можна робити зовні класу все, що завгодно. До захищених і приватних полів не можна звертатися ззовні класу, щоб не порушити цілісність даних класу.

Використовуючи інкапсуляцію, автор класу може захистити свої дані від некоректного використання. Крім того, вона замислювалася для полегшення сумісної розробки класів. Малося на увазі, що зміна способу зберігання даних, якщо вони оголошені як захищені або приватні не вимагає відповідних змін в класах, які використовують змінений клас. Наприклад, якщо в старій версії класу дані зберігалися у вигляді лінійного списку, а в новій версії - у вигляді дерева, ті класи, які були написані до зміни формату зберігання даних, переписувати не буде потрібно, якщо дані були приватними або захищеними (у останньому випадку - якщо використовуючі класи не були класами-нащадками), оскільки жоден з них цих класів не міг би безпосередньо звертатися до даних, а тільки через стандартні функції, які в новій версії мають вже коректно працювати з новим форматом даних.

Під час розгляду абстракції великі переваги дає виділення об'єктів з певною визначеною функціональністю. При аналізі проблемної області часто буває, що кілька об'єктів мають очевидно схожі структури чи функціональність. Якщо поведження об'єкта чи класу аналогічно поведженню іншого класу, то маємо справу з простим наслідуванням. Якщо ж у функціональності чи структурі класу присутні риси декількох класів, то має місце випадок множинного наслідування.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						7
Зм.	Арк.	№ докум.	Підп.	Дата		

Наслідкування часто пояснюють як можливість установити відношення "приналежності" між об'єктами, класами й типами даних. Яскравий приклад множинного спадкування: качконіс – успадковує характерні риси як ссавців, так і пернатих. Таким чином, ієрархія — це ще одна характеристика об'єктів, яку завжди варто відшукувати, займаючись об'єктно-орієнтованим аналізом і проектуванням.

При спадкуванні не успадковуються:

- 1 - Конструктори
- 2 - Деструктор
- 3 - Операція присвоювання

Спадкування властивостей і можливостей контролюються за допомогою кваліфікаторів доступу, які задаються при спадкуванні: `public`, `protected`, `private`.

Публічні функції може викликати будь-хто, а захищені і власні - тільки функції-члени і друзі.

Кваліфікатори доступу обмежують видимість повністю або частково для повністю або частково відкритих членів. Закриті члени завжди залишаються закритими. При спадкуванні можна зменшити видимість членів, але не розширити їх видимість.

У C++ об'єкти можна створювати також на основі структур і об'єднань

Структуру можна розглядати як праобраз поняття клас. Спочатку концепція структури (наприклад, у C) застосовувалася виключно для об'єднання різнорідних даних в єдиній конструкції для спільної обробки, тобто в понятті структури була присутня виключно інформаційна складова. У C++ у структурі можуть бути і функції-члени, у тому числі спеціальні члени – конструктори і деструктори. Структури, нарівні з класами, можна використовувати для повноцінного опису типів даних, що включає описи операцій, застосованих до описуваного типу даних. Однак наявність інформаційних членів у відкритій секції порушує один з основних принципів ООП – принцип інкапсуляції.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						8
Зм.	Арк.	№ докум.	Підп.	Дата		



Для доступу до внутрішніх інформаційних членів об'єкта, створеного на основі класу (читання / запис), необхідно використовувати спеціальні методи класу, звані модифікаторами (setters) і селекторами (getters). Вони здійснюють підконтрольне зчитування й зміну внутрішніх інформаційних членів.

### Конструктори й деструктори

Конструктори й деструктори є спеціальними методами класу. Конструктори викликаються при створенні об'єктів класу й відведенні пам'яті під них.

Конструктор без параметрів називають конструктором за замовчуванням. Такий конструктор зазвичай ініціалізує поля класу константними значеннями. Якщо конструктор для будь-якого класу не визначено, то компілятор сам

генерує конструктор за замовчуванням. Такі конструктори не присвоюють початкові значення полям класу. Тому, якщо треба однозначно ініціалізувати поля, треба визначити власний конструктор (ним може бути і конструктор за замовчуванням).

Деструктор – це особлива форма методу, який застосовується для звільнення пам'яті, зайнятої об'єктом. Деструктор за суттю є антиподом конструктора. Він викликається автоматично, коли об'єкт виходить з області видимості.

Деструктор не має аргументів і значення, яке повертається. Але він може виконувати деякі дії, наприклад, виведення остаточних значень елементів даних класу, що буває зручно при налагодженні програми.

Якщо деструктор явно не визначено, компілятор автоматично створює порожній деструктор.

Розміщувати деструктор у класі явно треба у разі, якщо об'єкт містить покажчики на пам'ять, яка виділяється динамічно, – інакше при знищенні об'єкта пам'ять, на яку посилались його поля-покажчики, не буде позначено як звільнену. Покажчик на деструктор визначити не можна. Деструктор не успадковується.

Так як конструктори й деструктори неявно входять в інтерфейс об'єкта, їх слід розташовувати у відкритій області класу.

Примітка. Конструктори й деструктори можуть розташовуватися й у закритій області для блокування можливості неявного створення об'єкта. Але в цьому випадку явне створення об'єкта можливо тільки при використанні статичних методів, які є частиною класу, а не конкретного об'єкта.

Відмінності і особливості опису конструктора від звичайної функції Ім'я конструктора співпадає з ім'ям класу.

При описі конструктора не вказується тип значення, яке повертається.

Слід зазначити, що звичайна процедура може не повертати значення, а тільки переробляти наявні дані. В цьому випадку при описі відповідної функції вказується спеціальний тип значення *void*.

В описі конструктора тип значення, що повертається, не вказується не тому, що значення, що повертається немає. Навпаки, воно є. Адже результатом роботи конструктора, відповідно до його назви, є створений об'єкт того типу, який описується даним класом. Страуструп відзначав, що конструктор – це те, що область пам'яті перетворює в об'єкт.

Конструктори можна класифікувати різними способами:

- за наявністю параметрів:

- 1) без параметрів,
- 2) з параметрами;

- за кількістю і типом параметрів:

- 1) конструктор за замовчуванням,
- 2) конструктор перетворення,
- 3) конструктор копіювання,
- 4) конструктор з двома і більше параметрами.

Набір і типи параметрів залежать від того, на основі яких даних створюється об'єкт.

### 1.3 Типові рішення ООП для баз даних

Об'єктно-орієнтована база даних (ООБД) — база даних, в якій дані моделюються у вигляді класів і об'єктів, їх атрибутів і методів.

Перші публікації про об'єктно-орієнтовані бази даних з'явилися всередині 1980-х років і були викликані проблемами роботи з даними, які виникають в системах збереження даних з високою складністю структур даних та відношень між ними.

Об'єктно-орієнтована база даних зазвичай рекомендована для тих випадків, коли потрібна високопродуктивна обробка даних, що мають складну структуру. У маніфесті ООБД пропонуються обов'язкові характеристики, котрим повинна відповідати будь-яка ООБД. Їх вибір засновано на 2-х критеріях: система має бути об'єктно-орієнтованою і бути базою даних.

Обов'язкові характеристики:

1. Підтримка складних об'єктів. У системі має бути передбачена можливість створення складових об'єктів за рахунок застосування конструкторів. Необхідно, щоб конструктори об'єктів були ортогональними, тобто будь-який з конструкторів можна було використовувати з будь-яким об'єктом.

2. Підтримка індивідуальності об'єктів. Усі об'єкти мають мати унікальний ідентифікатор, котрий не залежить від значень їх атрибутів.

3. Підтримка інкапсуляції. Коректна інкапсуляція досягається за рахунок того, що програмісти мають право доступу тільки до специфікації інтерфейсу методів, а дані й реалізація методів приховані всередині об'єктів.

4. Підтримка типів і класів. Вимагається, щоб в ООБД підтримувалась хоча б одна концепція відмінності між типами й класами. (Термін «тип» більш відповідає поняттю абстрактного типу даних. У мовах програмування змінна оголошується із вказівкою на її тип. Компілятор може використати цю інформацію для перевірки операцій, що виконуються зі змінною щодо сумісності з її типом, що дозволяє гарантувати коректність програмного забезпечення. З іншого боку, клас є шаблоном для створення об'єктів і надає методи, котрі можуть застосовуватися до цих об'єктів. Таким чином, поняття «клас» більше стосується часу виконання, аніж часу компіляції.)

5. Підтримка наслідування типів і класів від їх предків. Підтип або підклас має наслідувати атрибути й методи від, відповідно, його супертипу або суперкласу.

6. Перезавантаження в поєднанні з повним пов'язуванням. Методи повинні застосовуватися до об'єктів різних типів. Реалізація методу має залежати від типу об'єктів, до яких даний метод застосовується. Для забезпечення цієї функціональності зв'язування імен методів у системі не повинно виконуватись до часу, коли програму виконано.

7. Обчислювальна повнота. Мова маніпулювання даними має бути мовою програмування загального призначення.

8. Набір типів даних має бути розширюваним. Користувач повинен мати засоби створення нових типів даних на основі набору зумовлених системних типів. Більш того, між способами використання системних і користувацьких типів даних не повинно бути жодних відмінностей.

Необов'язкові характеристики:

- множинне спадкування;
- перевірка типів;
- розподілення;
- проектні транзакції.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						12
Зм.	Арк.	№ докум.	Підп.	Дата		

Відкриті характеристики:

- парадигми програмування (процедурне, декларативне);
- система представлення;
- система типів;
- одноманітність. Реалізація — мова програмування — інтерфейс.

Результатом суміщення можливостей (особливостей) баз даних і можливостей об'єктно-орієнтованих мов програмування є об'єктно-орієнтовані системи керування базами даних (ООСКБД). ООСКБД дозволяє працювати з об'єктами баз даних так само, як з об'єктами в програмуванні в ООМП. ООСКБД розширює мови програмування, прозоро додаючи довгочасні дані, керування паралелізмом, поновлення даних, асоційовані запити та інші можливості.

Деякі об'єктно-орієнтовані бази даних розроблені для щільної взаємодії з такими об'єктно-орієнтованими мовами програмування як Python, Java, C#, Visual Basic .NET, C++, Objective-C і Smalltalk; інші мають свої власні мови програмування. ООСКБД використовують таку ж модель, що й об'єктно-орієнтовані мови програмування.

СУБД повинна забезпечувати:

- довготермінове зберігання;
- використання зовнішньої пам'яті;
- паралелізм;
- відновлення;
- нерегламентовані запити.

## 2. РОЗРОБКА ПРОГРАМИ ВИКОНАННЯ ЗАВДАННЯ

### 2.1 Аналіз предметної області “Розробники комп’ютерних ігор”

Розробники комп’ютерних ігор – це онлайн база даних про організації, які займаються створенням ігор, і інформація про створені ними продукти. У роботі сайту можна виділити ряд ситуацій:

#### 1. Організація пошуку продукту або організації за заданими критеріями.

Необхідно вести загальну інформацію про уже відомі компоненти. Інформація організації включає: дату створення, розміщення, засновники, чим є особливою, які жанри зазвичай створює. Інформація про створений продукт містить: дату створення, жанр, тип гри (2D, 3D, інді і т.п.), чи є можливість онлайн або мультиплеєрної гри, ціна і можливість придбання, платформи, на яких вона доступна і можливість пошуку за одним із цих критеріїв.

#### 2. Введення нових компонентів (організацій/продуктів).

Після створення нової гри однією із уже введених організацій, потрібно додати цей продукт і інформацію про нього згідно у базу із стандартом, згаданим вище, цим займається один із адміністраторів сервера. Завдяки людині, яка відповідальна за усі оновлення і інформацію у сфері комп’ютерних ігор, сайт буде постійно оновлюватись і завжди залишатиметься «найсвіжішим». На початку введення нового продукту, адміністратор здійснює пошук товару на наявність його в базі. Якщо його немає в базі, необхідно знайти потрібну організацію і внести нові дані створеного нею продукту або просто додати нову організацію в базу.

#### 3. Контакт із клієнтами.

Із можливостей передбачено співробітництво із гостем. Окрім того, що він зможе переглянути потрібну йому інформацію щодо організації або її продукту, у нього буде також можливість запропонувати свою інформацію або корективи без реєстрації. Уся запропонована інформація буде зберігатися в окремому файлі, до якого матимуть доступ тільки адміністратори і, перевібивши інформацію і корективи, вони зможуть додати це у базу даних.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						14
Зм.	Арк.	№ докум.	Підп.	Дата		

## 2.2 Постановка завдання

Створити консольну базу даних на C++ по предметній області «Розробники комп'ютерних ігор» за зразком бази даних Access Борей. Необхідно реалізувати такі функції:

1. Додавання запису у базу.
2. Видалення запису з бази.
3. Редагування запису у базі.
4. Формування списку на основі первинних документів.
5. Запити на виведення даних, що відповідають певним умовам.
6. Форматований вивід списку.
7. Відбір та групування списку унікальних даних на основі загального списку даних.
8. Сортування списку.

## 2.3 Вхідні дані

Вхідними даними є дані, які вводяться адміністратором, а саме список організацій, список її продуктів (ігор) і інформація по заданим критеріям, список адміністраторів і діяльність, за яку вони відповідальні, а також загальне поняття про сайт і його можливості, які були описані вище. Адміністратор має мати змогу додавати інформацію про організацію або її продукт, редагувати уже наявну інформацію і перевіряти файл із запропонованою інформацією або корективами. Також він має змогу переглянути уже наявну інформацію у базі даних.

## 2.4 Розробка моделі бази даних

На основі результатів аналізу предметної області та вхідної інформації виділено елементи, які необхідні у базі даних. Вирішено прийняти модель бази даних із зберіганням блоків даних у окремих таблицях. Таблицями для зберігання даних слугують: таблиця даних про організації, таблиця даних про їх продукти,

таблиця даних про адміністраторів, таблиця даних про запропоновані корективи.

Додавання запису у базу.

1. Видалення запису з бази.
2. Редагування запису у базі.
3. Формування списку на основі первинних документів.
4. Запити на виведення даних, що відповідають певним умовам.
5. Форматований вивід списку.
6. Відбір та групування списку унікальних даних на основі загального списку даних.
7. Сортування списку.

## 2.5 Вхідні дані

Вхідними даними є дані, які вводяться користувачем (працівником магазину), а саме список книжок, список клієнтів, список працівників, а також загальне поняття про бізнес-логіку роботи книжкового магазину, описане вище.

Працівник має мати змогу додавати запис про книгу, редагувати кількість примірників при надходженні книги на склад, видаляти запис про товар. Також він має змогу переглянути весь товар, зареєстрований на складі.

## 2.6 Розробка моделі бази даних

На основі результатів аналізу предметної області та вхідної інформації виділено елементи, які необхідні у базі даних. Вирішено прийняти модель бази даних із зберіганням блоків даних у окремих таблицях. Таблицями для зберігання даних слугують: таблиця даних про організації, таблиця даних про їх продукти, таблиця даних про адміністраторів, таблиця даних про запропоновані корективи. Структура таблиць даних наведена нижче у табл. 2.1-2.3.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						16
Зм.	Арк.	№ докум.	Підп.	Дата		



Таблиця 2.1. Структура таблиці бази даних «Організації»

Organization (назва поля)	Тип даних	Опис
Organization_code	int	Унікальний код організації, має генеруватися автоматично
publ_year	string	Рік, коли була заснована, вводиться з консолі у вигляді YYYY
Organization_name	string	Назва організації (поточна), вводиться з консолі рядок слів, розділених пробілами
Organization_place	string	Інформація, де розташована, вводиться з консолі
Organization_genre	string	Жанр ігор, на якому спеціалізується, вводиться з консолі
Organization_platform	string	Платформи, для яких випускає ігри, вводиться з консолі
Organization_establisher	string	Засновники організації (ім'я та прізвище), вводиться з консолі рядок слів, розділених пробілами
Organization_info	string	Загальна інформація про організацію, вводиться з консолі
Organization_games	string	Найпопулярніші видання організації, вводиться з консолі

Таблицю «Організації» (Organization) вирішено занести у файл "Organization.txt".

Таблиця 2.2. Структура таблиці бази даних «Продукти»

Organization (назва поля)	Тип даних	Опис
Game_code	int	Унікальний код організації, має генеруватися автоматично
Game_publ_year	string	Рік, коли була заснована, вводиться з консолі у вигляді YYYY
Game_name	string	Назва ігри (поточна), вводиться з консолі рядок слів, розділених пробілами
Game_platform	string	Інформація, на яких платформах доступна, вводиться з консолі рядок слів, розділених пробілами
Game_genre	string	Жанр ігри, вводиться з консолі
Game_info	string	Короткий опис гри, вводиться з консолі
Game_type	string	Типу ігри, вводиться з консолі
Game_online	string	Чи є можливість онлайн або мультиплеєрної гри, вводиться з консолі рядок слів, розділених пробілами
Game_price	string	Ціна гри (у гривнях), вводиться з консолі
Game_organization	string	Назва організації, яка створила продукт, вводиться з консолі рядок слів, розділених пробілами

Таблицю «Продукти» (Products) вирішено занести у файл “Products.txt”.

Таблиця 2.3. Структура таблиці бази даних «Адміністратори»

Organization (назва поля)	Тип даних	Опис
Admin_code	int	Унікальний код організації, має генеруватися автоматично
Date_start	string	Дата прийняття на роботу, вводиться з консолі
DateFinish	string	Дата звільнення, вводиться з консолі
Admin_type	string	За який тип роботи відповідальний (база даних, поповнення' інформації і т.п.), вводиться з консолі у рядок, розділених пробілами
Admin_name	string	Ім'я, вводиться з консолі
Admin_surname	string	Прізвище, вводиться з консолі
Admin_middlename	string	По-батькові, вводиться з консолі
Admin_Phone	string	Номер телефону, вводиться з консолі
Admin_settlement	string	Місце проживання, вводиться з консолі
Admin_house	string	Будинок, вводиться з консолі
Admin_flat	string	Квартира, вводиться з консолі
Admin_street	string	Вулиця, вводиться з консолі
Admin_post	string	Поштовий код, вводиться з консолі
Admin_email	string	Електронна адреса, вводиться з консолі

Таблицю «Адміністратори» (Admins) вирішено занести у файл “Admins.txt”.

Таблиця 2.4. Структура таблиці бази даних «Запропоноване»

Organization (назва поля)	Тип даних	Опис
Sug_date	int	Дата введення, автоматично записується
Sug_text	string	Увесь текст, введений гостем, автоматично записується
Sug_topic	string	Тема, введена гостем, автоматично записується
Sug_code	int	Код запропонованого топіку, генерується автоматично

Таблицю «Запропоноване» (Suggestion) вирішено занести у файл “ Suggestion.txt”.

Для реалізації завдання з врахуванням результатів аналізу предметної області було створено об’єктно-класову модель. Модель описано таблицями та діаграмою класів, наведеними у додатку В.

## 2.5 Розробка алгоритму вирішення задач, що виникли під час виконання роботи

### 2.5.1 Додавання запису в базу

#### Додавання нової організації

Генерування коду відбувається під час додавання даних про організацію, перевіркою наявності файлу Organizations\_file.txt, при його наявності, відбувається його читання. Дійшовши до кінця файлу, код генерується таким чином: останній наявний код + 1. Якщо файл пустий і не містить жодної інформації, код автоматично починається із «1000» і записується у файл для першої організації. Якщо файлу немає, то він автоматично створюється.

Метод додавання даних про організацію (AddOrganization()) вміщує у собі функцію генерації ідентифікаційного коду організації, при записі даних у файл.

З клавіатури вводиться назва, місце розташування, найпопулярніший жанр продукту, назви платформ, на яких випускає свої продукти, ППІ засновника організації (може бути декілька), ігри, які випустила, загальна інформація, рік заснування. Код автоматично генерується при записі даних у файл. Ці дані виводяться в консоль для їх огляду і підтвердження правильності вводу, після чого записуються у файл.

Додавання нового продукту або адміністратора здійснюється аналогічно у класах `games` та `administrators` з використанням змінних-членів відповідних класів.

Для реалізації завдання з врахуванням результатів аналізу предметної області було створено об'єктно-класову модель. Модель описано таблицями та діаграмою класів, наведеними у додатку В.

### 2.5.2 Видалення запису з бази

Видалення реалізовано лише для стрічки даних з файлу. Розглянемо цей алгоритм на прикладі класу «`administrators`». Метод видалення адміністратора (`delete administrators()`): виконується пошук адміністратора за попередньо-введеним ідентифікаційним кодом, якщо його знайдено, рядок із усією інформацією про нього пропускається, при продовженні, наступні ідентифікаційні коди адміністраторів зменшуються на 1, якщо не знайдено – то просто переписує рядок у проміжну змінну. Після завершення видалення змінені дані перезаписуються у файл.

### 2.5.3 Редагування запису у базі

Редагування створене для стрічок даних з файлів “Organizations\_file.txt”, “Administrators\_file.txt” і “games\_file.txt”. Розглянемо цей алгоритм на прикладі класу «organizations».

Метод редагування даних про організацію (edit\_organizations()): з клавіатури вводиться код організації, дані якої потрібно змінити. Відкривається файл «Organizations\_file» і по одному першому слову (коду) кожного рядку зчитуються дані, при знаходженні потрібного коду, виводиться інформація про організацію і система запитує, що саме потрібно змінити, при підтвердженні нововведених змін, із початку файлу по одному першому слову (коду) кожного рядку знову зчитуються дані, якщо код не збігається з введеним – рядок записується в проміжну змінну, якщо збігається – заміняє необхідну інформацію про організацію і записується в змінну. Після завершення редагування змінені дані перезаписуються у файл.

### 2.5.4 Виведення інформації з бази

Виведення інформації з бази даних створене для усіх файлів. Розглянемо цей алгоритм на прикладі класу «games».

Метод виведення інформації про гру (Gread()): з клавіатури вводиться спосіб пошуку потрібної гри: 1 – через ідентифікаційний код, 2 – вивести першу гру у списку, після чого, можна вивести також і наступні так само. Через введення ідентифікаційного коду, відкривається файл «games\_file.txt», у якому, перевіряється перше слово (код), якщо він ідентичний введеному – то виводиться стрічка інформації про гру у певному порядку.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						25
Зм.	Арк.	№ докум.	Підп.	Дата		

					КР.ІП-25.00.00.000 ПЗ	Арк.
						26
Зм.	Арк.	№ докум.	Підп.	Дата		



					КР.ІП-25.00.00.000 ПЗ	Арк.
						27
Зм.	Арк.	№ докум.	Підп.	Дата		

					КР.ІП-25.00.00.000 ПЗ	Арк.
						28
Зм.	Арк.	№ докум.	Підп.	Дата		

### 3. КОНТРОЛЬНИЙ ПРИКЛАД

Нижче наведено опис ситуацій та вхідні дані, які при цьому виникають, та інформація, яку необхідно отримати з бази даних. Реалізація ситуацій з використанням розробленого програмного забезпечення проілюстрована у вигляді скріншотів, які наведено у додатку Б. Посилання на конкретний рисунок можна знайти у тексті опису кожної ситуації.

У програмі включено бібліотеки <locale>, "windows.h", функції SetConsoleCP(1251), SetConsoleOutputCP(1251).

Слід враховувати, що під час виконання контрольного прикладу було змінено дані в базі, тому повторне виконання контрольного прикладу приведе до інших результатів з врахуванням проведених змін.

#### **Ситуація 1: додавання інформації**

Надійшла інформація про нову організацію і її продукт (ігру).

№1: Nintendo, 1889 року, Японія, платформери, консоль і пк, Фусадзіро Ямауті, Super Mario Bros і Zelda, організація яка розпочинала із створення гральних карт і продовжує зараз створювати світові шедеври.

№2: Super Mario Bros, 1985 року, платформер, консоль і пк, 2D, 1199 грн, Nintendo, Занесена до «Книги рекордів Гіннеса як гра, що розійшлась найбільшим тиражем у історії.

Адміністратор, який відповідальний за внесення інформації у бази даних, повинен виконати наступні дії:

Вибрати розділ «Організації» у пункті меню «Переглянути вміст або інформацію».

Переконатися у наявності чи відсутності інформації про таку організацію у базі даних, виконавши пошук, який є окремим пунктом меню (Рис. Б.1).

					КР.ІП-25.00.00.000 ПЗ	Арк.
						29
Зм.	Арк.	№ докум.	Підп.	Дата		

Якщо організацію знайдено, тоді не потрібно добавляти інформацію про організацію.

Якщо організацію не знайдено, потрібно додати інформацію про неї, вибравши «База даних організацій» у пункті меню «Додати дані в базу даних».

Виконати запис необхідної інформації про організацію і підтвердити додавання її у базу.

Виконати такі ж операції із продуктом організації, а саме:

Переконатися у наявності чи відсутності інформації про таку організацію у базі даних, виконавши пошук, для чого потрібно обрати «База даних продуктів» у пункті меню «Переглянути вміст або інформацію».

Якщо продукт знайдено, не добавляти інформацію про нього.

Якщо продукту не знайдено, додати його у базу даних, обравши «База даних продуктів» у пункті меню «Додати дані в базу даних» і заповнивши необхідну інформацію про гру, після чого підтвердити ввід інформації у базу.

На цьому ситуацію 1 закінчено, для повернення в головне меню потрібно обрати функцію «Повернутись у головне меню».

## Ситуація 2: редагування інформації

Розглянемо ситуацію, коли інформація про організацію невірна і її потрібно змінити.

Blizzard Entertainment, 1991 року, Ірвайн, стратегії, ПК, Майкл Морхейм і Френк Пірс і Аллен Адам, warcraft і diablo, організація із найпопулярнішою стратегією за усі часи.

Потрібно змінити рік заснування організації, із невірного, старого – 1990 – на правильний - 1991

1. Для зміни інформації, потрібно обрати пункт «Редагувати бази даних» у головному меню, після обрати «Редагувати базу даних організацій» і ввести ідентифікаційний код організації.

					КР.ІП-25.00.00.000 ПЗ	Арк.
						28
Зм.	Арк.	№ докум.	Підп.	Дата		

2. Для зміни інформації, потрібно обрати пункт «Редагувати бази даних» у головному меню, після обрати «Редагувати базу даних організацій» і ввести ідентифікаційний код організації.
3. Відбувається пошук потрібної організації. Коли організацію знайдено, виводиться інформація про неї і система запитує що саме потрібно змінити, обираємо «Рік заснування організації»
4. Наступний крок – ввести новий рік заснування організації.
5. Після введення нового року, система запитує, чи ми правильно все ввели, підтвердивши введення, система перезапише інформацію.
6. Якщо організації не було знайдено, система виведе помилку і попросить ще раз ввести код.
7. На цьому ситуацію 2 закінчено, для повернення в головне меню потрібно обрати функцію «Повернення у головне меню».

### **Ситуація 3: перегляд бази даних**

Для того, щоб в загальному поглянути на базу про організації, потрібно виконати такі дії:

1. Вибрати розділ «Переглянути вміст або інформацію» у пункті головного меню, після чого, обрати базу «Організацій».
2. Вивести потрібну організацію за ідентифікаційним кодом або по порядку кожну організацію у списку (Рис. Б.3).
3. На цьому ситуацію 3 закінчено, для повернення в головне меню потрібно обрати функцію «Повернення у головне меню».

### **Ситуація 4: приймання на роботу нового працівника**

На роботу наймається Галим Роман Романович, телефон: +380963554875, e-mail: [gal\\_roman@ukr.net](mailto:gal_roman@ukr.net), адреса: Україна, Галич, вул. Шевченка 4, 12567, на посаду інформаційного адміністратора,

					КР.ІП-25.00.00.000 ПЗ	Арк.
						30
Зм.	Арк.	№ докум.	Підп.	Дата		

1. Вибрати розділ «Додати дані в базу даних» у головному меню, після чого обрати базу даних «Адміністраторів».
2. Заповнити відповідні поля (інформацію) про адміністратора (Рис. Б.4).
3. На цьому ситуацію 4 закінчено, для повернення в головне меню потрібно обрати функцію «Повернутись у головне меню».

#### **Ситуація 5: звільнення адміністратора**

Потрібно змінити дані про працівника Кабалова Івана Назаровича, звільнений 03/10/2019.

1. Вибрати розділ «Редагувати бази даних» у головному меню, після чого обрати «Редагувати базу даних адміністраторів».
2. Ввести ідентифікаційний код адміністратора, після чого відбудеться його пошук у базі даних (Рис. Б.5).
3. Обрати «Змінити дату звільнення» і змінити дату, коли працівник був звільнений (Рис. Б.5).
4. На цьому ситуацію 5 закінчено, для повернення в головне меню потрібно обрати «Повернутись у головне меню».

#### **Ситуація 6: зміна даних існуючого продавця**

Потрібно змінити дані адміністратора у відношенні до компанії про робочого Саматор Марію Іванівну, стала адміністратором інформаційної сфери.

1. Відбувається пошук працівника за ідентифікаційним кодом (Рис. Б.6).
2. Введення змін і їх підтвердження, після чого відбувається перезапис інформації (Рис. Б.6).
3. На цьому ситуацію 6 закінчено, для повернення в головне меню потрібно обрати функцію «Повернутись у головне меню».

					КР.ІП-25.00.00.000 ПЗ	Арк.
						31
Зм.	Арк.	№ докум.	Підп.	Дата		

### Ситуація 7: видалення інформації про продукт

Потрібно видалити усю інформацію про певний продукт (гру), для цього необхідно:

1. Вибрати розділ «Видалити інформацію з бази даних» у головному меню, після чого обрати «База даних продуктів» (Рис. Б.7).
2. Ввести ідентифікаційний код продукту і підтвердити видалення інформації про нього (Рис. Б.7).
3. На цьому ситуацію 7 закінчено, для повернення в головне меню потрібно обрати функцію «Повернутись у головне меню».

					КР.ІП-25.00.00.000 ПЗ	Арк.
						32
Зм.	Арк.	№ докум.	Підп.	Дата		

## ВИСНОВКИ

Виконання курсової роботи була із метою освоєння усіх теоретичних знань, отриманих з дисципліни «Об'єктно-орієнтоване програмування» і використання їх на практиці програмування мовою C++.

У цій курсовій роботі було зроблено задачу розробки проекту консольної бази даних по предметній області «Розробники комп'ютерних ігор». У ній передбачалася робота з функціями та файлами, виділенням пам'яті, алгоритмом пошуку унікальних значень і т. п.

Програму було написано у середовищі Visual Studio – C++. Перевірка роботи програмі і її тестування показали, що завдання реалізовано коректно.

Розроблена система використання баз даних і їх управління, завдяки чому, можна повноцінно реалізовувати бази даних і можливі маніпуляції з ними, завдяки основних розроблених функцій, які надають можливість роботи з базами даних і навіть можливість використання власних методів.

Під час написання курсової роботи освоювались основні бібліотеки мови C++ і їх засоби реалізації, а також були поглиблені знання про роботу з консоллю і базами.

Вивчивши та проаналізувавши мову C++, зроблено висновок, що ця мова є досить гнучким і доступним інструментом для роботи з базами даних і також використання у проєктах, що реалізовані на різних мовах програмування.

Отже, можна зробити висновок, що C++ - це мова програмування високого рівня, яка попри виконання інших завдань, підходить і для використання бази даних.



## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Юрчишин В.М. Програмування: навч. посіб. / В. М. Юрчишин, Б. В. Клим, В. Б. Кропивницька. - Івано-Франківськ: ІФНТУНГ, 2012. - 188 с.
2. Гарасимів Т. Г. Структурне програмування: лабораторний практикум / Т. Г. Гарасимів, В. Б. Кропивницька, В. М. Гарасимів - Івано-Франківськ: ІФНТУНГ, 2016. – 117 с.
3. Безчнюк Ю. В. Сучасні технології програмування: конспект лекцій / Ю. В. Безчачнюк, Л. О. Штаєр. - Івано-Франківськ: ІФНТУНГ, 2011. - 161 с.
4. Ковалюк Т. В. Основи програмування. Теорія та практика: О.Г. Трофименко, Ю. В. Прокоп, І. Г. Швайко, Л. М. Буката - Одеса: Фенікс 2010. - 544 с.
5. Інтернет-джерело «Вікіпедія».
6. Ковалюк Т. В. Основи програмування: підручник. - К.: Видавнича група BHV, 2005. - 384 с.: іл. - (Інформатика). – 378

					КР.ІП-25.00.00.000 ПЗ	Арк.
						34
Зм.	Арк.	№ докум.	Підп.	Дата		

# Додатки

					КР.ІП-25.00.00.000 ПЗ							
Змн.	Лист	№ докум.	Підпис	Дата								
Розроб.		Корнута Ю. В.			Розробка програмного рішення для консольної бази даних по предметній області «Книжковий магазин» засобами C++				Літ.	Арк.	Акрушів	
Перевір.		Шекета В. І.									35	103
Реценз.									ІФНТУНГ, ІП-17-2			
Н. Контр.												
Затверд.												

## Додаток А

### Код програми мовою C++

#### Class organizations

```
1. #include <iostream>
2. #include <string>
3. #include <fstream>
4. #include <locale>
5. #include "windows.h"
6. using namespace std;
7. class organizations{
8. private:
9. int organization_code = 0;
10. string publ_year, organization_name, organization_place,
    organization_genre, organization_platform,
11. organization_establisher, organization_games, organization_info,
    path = "Organizations_file.txt";
12. public:
13. void delete_organization(int choise, string temp, int itemp,
    string str){
14. ifstream organizations_file(path, ios::binary);
15. Owrong_code:
16. organizations_file.clear();
17. organizations_file.seekg(0, ios_base::beg);
18. cout << "Введіть ідентифікаційний код організації, інформацію
    про яку хочете видалити:" << endl;
19. cin >> temp;
20. choise = atoi(temp.c_str());;
21. itemp = choise;
22. organizations_file.clear();
23. while (!organizations_file.eof()){
24. Ostart:
25. if (organizations_file.eof()){
26. cout << "\n\n\nОрганізації із таким ідентифікаційним кодом
    неіснує\n\n\n";
27. goto Owrong_code;}
28. organizations_file >> temp;
29. if (atoi(temp.c_str()) == choise){
30. cout << "\n\n\nОрганізація із ідентифікаційним кодом: " <<
    atoi(temp.c_str()) << endl;
31. cout << "\nНазва організації:\n";
32. choise = 0;}
33. else{
34. getline(organizations_file, temp);
35. goto Ostart;}
36. while (temp != "|"){
37. temp = "";
38. organizations_file >> temp;
39. if (temp == "|"){}
}
```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						36
Зм.	Арк.	№ докум.	Підп.	Дата		

```

40. else{
41. cout << temp << " ";}}
42. break;}
43. cout << "\n\n\nВи впевненні, що хочете видалити інформацію про
    неї?\n\nВведіть:\n1 - Якщо ні\nБудь-що інше - Якщо так\n";
44. cin >> temp;
45. cin.ignore();
46. choise = atoi(temp.c_str());;
47. if (choise == 1){
48. goto Owrong_code; }
49. choise = itemp;
50. organizations_file.clear();
51. organizations_file.seekg(0, ios_base::beg);
52. while (!organizations_file.eof()){
53. Gend:
54. organizations_file >> temp;
55. if (size(temp) != 4){
56. break;}
57. if (itemp == 1){
58. choise = atoi(temp.c_str()) - 1;
59. temp = to_string(choise);
60. str += temp;
61. getline(organizations_file, temp);
62. str += temp + "\n";
63. goto Gend;}
64. if (atoi(temp.c_str()) == choise){
65. getline(organizations_file, temp);
66. itemp = 1;}
67. else{
68. str += temp;
69. getline(organizations_file, temp);
70. str += temp + "\n";}}
71. ofstream Oout(path, ios::binary);
72. Oout << str;
73. cout << "\n\n\nЗміни успішно збережено!\n\n\n";
74. return;}
75. void edit_organizations(int choise, string temp, int itemp,
    string new_word, string str){
76. int i = 0, k = 1;
77. ifstream organizations_file(path, ios::binary);
78. Gwrong_code:
79. organizations_file.clear();
80. organizations_file.seekg(0, ios_base::beg);
81. cout << "Введіть ідентифікаційний код організації, інформацію
    про яку хочете змінити:" << endl;
82. cin >> temp;
83. choise = atoi(temp.c_str());;
84. itemp = choise;
85. organizations_file.clear();
86. while (!organizations_file.eof()){
87. start1:
88. if (organizations_file.eof()){
89. cout << "\n\n\nОрганізації із таким ідентифікаційним кодом
    неіснує\n\n\n";

```

Зм.	Арк.	№ докум.	Підп.	Дата

**КР.ІП-25.00.00.000 ПЗ**

Арк.

37

```

90. goto Gwrong_code;
91. break;}
92. organizations_file >> temp;
93. if (atoi(temp.c_str()) == choise){
94. cout << "Організація із ідентифікаційним кодом: " <<
    atoi(temp.c_str()) << endl;
95. cout << "\nНазва організації:\n";
96. choise = 1;}
97. else{
98. getline(organizations_file, temp);
99. goto start1;}
100. Ggain:
101. temp = "";
102. while (temp != "|"){
103. organizations_file >> temp;
104. if (temp == "|"){
105. else{
106. cout << temp << " ";}}
107. choise++;
108. switch (choise){
109. case 2:{
110. cout << "\n";
111. break;}
112. case 3:{
113. cout << "року заснування\nМісце розташування:\n";
114. break;}
115. case 4:{
116. cout << "\nНайпопулярніші жанри ігор:\n";
117. break;}
118. case 5:{
119. cout << "\nНайпопулярніші платформи для створених ігор:\n";
120. break;}
121. case 6:{
122. cout << "\nЗасновник/и організації:\n";
123. break;}
124. case 7:{
125. cout << "\nНайпопулярніші ігри організації:\n";
126. break;}
127. case 8:{
128. cout << "\nЗагальна інформація про організацію:\n";
129. break;}}
130. if (choise < 9){
131. goto Ggain;}
132. else{
133. break;}}
134. Oedit_again:
135. cout << "\n\nВведіть: \n1 - Щоб змінити назву організації \n2
    - Щоб змінити рік заснування організації \n3 - Щоб змінити місце
    розташування організації \n4 - Щоб змінити найпопулярніші жанри
    ігор організації \n5 - Щоб змінити найпопулярніші платформи для
    створених ігор\n6 - Щоб змінити засновників організації \n7 -
    Щоб змінити найпопулярніші ігри організації \n8 - Щоб змінити
    загальну інформацію про організацію \n9 - Щоб повернутись у меню"
    << endl;

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						38
Зм.	Арк.	№ докум.	Підп.	Дата		

```

136. cin >> temp;
137. cin.ignore();
138. if (choise < 1){
139. cout << "\n\nНекоректний вибір, спробуйте знову";
140. goto Oedit_again;}
141. else
142. if (choise > 9){
143. cout << "\n\nНекоректний вибір, спробуйте знову";
144. goto Oedit_again;}
145. switch (choise){
146. case 1:
147. cout << "Введіть нову назву організації:" << endl;
148. getline(cin, temp);
149. break;}
150. case 2:{
151. cout << "Введіть новий рік заснування організації:" << endl;
152. getline(cin, temp);
153. break;}
154. case 3:{
155. cout << "Введіть нове місце розташування організації:" << endl;
156. getline(cin, temp);
157. break;}
158. case 4:{
159. cout << "Введіть нові найпопулярніші жанри ігор:" << endl;
160. getline(cin, temp);
161. break;}
162. case 5:{
163. cout << "Введіть найпопулярніші платформи для створених ігор:"
    << endl;
164. getline(cin, temp);
165. break;}
166. case 6:{
167. cout << "Введіть засновників організації(якщо їх декілька,
    розділяйте їх, використовуючи *і* або *та*):" << endl;
168. getline(cin, temp);
169. break;}
170. case 7:{
171. cout << "Введіть найпопулярніші ігри організації:" << endl;
172. getline(cin, temp);
173. break;}
174. case 8:{
175. cout << "Введіть загальну інформацію про організацію:" << endl;
176. getline(cin, temp);
177. break;}
178. case 9:{
179. return;}}
180. i = choise;
181. new_word = temp;
182. cout << "\n\nВи ввели:\n" << temp << "\n\nВведіть 1 - Щоб
    змінити 2 - Якщо усе введено правильно і щоб перезаписати у
    файл\n";
183. cin >> temp;
184. if (atoi(temp.c_str()) == 1){
185. goto Oedit_again;}
186. choise = itemp;
187. organizations_file.clear();

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						39
Зм.	Арк.	№ докум.	Підп.	Дата		

```

188. organizations_file.seekg(0, ios_base::beg);
189. while (!organizations_file.eof()){
190. organizations_file >> temp;
191. if (size(temp) != 4){
192. return;}
193. str += temp;
194. if (atoi(temp.c_str()) == choise){
195. for (k = 1; k < i; k++) {
196. getline(organizations_file, temp, '|');
197. str += temp + "|";}
198. if (k = i){
199. getline(organizations_file, temp, '|');
200. getline(organizations_file, temp);
201. str += " " + new_word + " |" + temp;}}
202. else {
203. getline(organizations_file, temp);
204. str += temp;
205. k = 1;}
206. str += "\n";}
207. ofstream Oout(path, ios::binary);
208. out << str;
209. cout << "\n\nЗміни успішно збережено!\n\n";
210. return;};
211. void AddOrganization(int choise, string temp){
212. cout << "Ви у функції додавання організації в базу даних" <<
endl;
213. fstream organizations_file(path, ios::in | ios::out | ios::app);
214. try{
215. organizations_file.open(path);
216. allagain:
217. organizations_file.clear();
218. cout << "Введіть назву організації" << endl;
219. getline(cin, organization_name);
220. cout << "Ви ввели: " << organization_name << endl;
221. cout << "Введіть місце розташування організації" << endl;
222. getline(cin, organization_place);
223. cout << "Ви ввели: " << organization_place << endl;
224. cout << "Введіть найпопулярніший жанр продукту організації" <<
endl;
225. getline(cin, organization_genre);
226. cout << "Ви ввели: " << organization_genre << endl;
227. cout << "Введіть назви платформ, на якій випускає свій продукт
організація" << endl;
228. getline(cin, organization_platform);
229. cout << "Ви ввели: " << organization_platform << endl;
230. cout << "Введіть ПІП засновника організації (якщо їх декілька,
розділяйте їх, використовуючи *і* або *та*):" << endl;
231. getline(cin, organization_establisher);
232. cout << "Ви ввели: " << organization_establisher << endl;
233. cout << "Введіть ігри, які випустила організація" << endl;
234. getline(cin, organization_games);
235. cout << "Ви ввели: " << organization_games << endl;
236. cout << "Введіть загальну інформацію про організацію" << endl;
237. getline(cin, organization_info);

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						40
Зм.	Арк.	№ докум.	Підп.	Дата		

```

238. cout << "Ви ввели: " << organization_info << endl;
239. cout << "Введіть рік (число) заснування організації" << endl;
240. cin >> publ_year;
241. cout << "Ви ввели: " << publ_year << endl;
242. again:
243. cout << "Назва організації: " << organization_name << "\nРік
заснування організації:" << publ_year << "\nмісце розташування: "
<< organization_place << "\nнайпопулярніший жанр продукту: " <<
organization_genre << "\nплатформи, на яких випускає свій продукт:
" << organization_platform << "\nзасновник організації: " <<
organization_establisher << "\nігри, які випустила: " <<
organization_games << "\nзагальна інформація: " <<
organization_info << "\nЧи усе правильно введено і можна вносити
дані у файл?\nВведіть: \n1 - Щоб змінити Назву організації \n2 -
Щоб змінити Місце розташування \n3 - Щоб змінити Найпопулярніший
жанр продукту \n4 - Щоб змінити Платформи, на яких випускає свій
продукт \n5 - Щоб змінити Засновника організації \n6 - Щоб змінити
Ігри, які випустила \n7 - Щоб змінити Загальну інформацію \n8 -
Щоб змінити Загальну інформацію \n9 - Щоб внести усі дані заново
\n0 - Якщо усі дані введено вірно " << endl;
244. cin >> temp;
245. choise = atoi(temp.c_str());
246. switch (choise){
247. case 1:
248. {cout << "Введіть назву організації" << endl;
249. cin.ignore();
250. getline(cin, organization_name);
251. cout << "Ви ввели: " << organization_name << endl;
252. break; }
253. case 2:{
254. cout << "Введіть місце розташування організації" << endl;
255. cin.ignore();
256. getline(cin, organization_place);
257. cout << "Ви ввели: " << organization_place << endl;
258. break; }
259. case 3:{
260. cout << "Введіть найпопулярніший жанр продукту організації" <<
endl;
261. cin.ignore();
262. getline(cin, organization_genre);
263. cout << "Ви ввели: " << organization_genre << endl;
264. break; }
265. case 4:{
266. cout << "Введіть назви платформ, на якій випускає свій продукт
організація" << endl;
267. cin.ignore();
268. getline(cin, organization_platform);
269. cout << "Ви ввели: " << organization_platform << endl;
270. break; }
271. case 5:{
272. cout << "Введіть ПІП засновника організації (якщо їх декілька -
то декілька)" << endl;
273. cin.ignore();
274. getline(cin, organization_establisher);

```



```

275. cout << "Ви ввели: " << organization_establisher << endl;
276. break; }
277. case 6:{
278. cout << "Введіть ігри, які випустила організація" << endl;
279. cin.ignore();
280. getline(cin, organization_games);
281. cout << "Ви ввели: " << organization_games << endl;
282. break; }
283. case 7:{
284. cout << "Введіть загальну інформацію про організацію" << endl;
285. cin.ignore();
286. getline(cin, organization_info);
287. cout << "Ви ввели: " << organization_info << endl;
288. break; }
289. case 8:{
290. cout << "Введіть рік (число) заснування організації" << endl;
291. cin.ignore();
292. cin >> publ_year;
293. cout << "Ви ввели: " << publ_year << endl;}
294. case 9: {
295. cin.ignore();
296. goto allagain;}
297. case 0: {break;}}
298. if ((choise < 9) && (choise > 0)){
299. cout << "Все правильно введено? Введіть: \n1 - якщо так \n2 -
якщо ні або потрібно ще щось змінити" << endl;
300. cin >> choise;
301. if (choise == 2) {
302. goto again;}}
303. temp = "";
304. organizations_file.clear();
305. string temp1;
306. while (!organizations_file.eof()){
307. organizations_file >> temp;
308. getline(organizations_file, temp1);}
309. if (temp == ""){
310. organization_code = 1000;}
311. else{
312. organization_code = atoi(temp.c_str());
313. organization_code++;}
314. organizations_file.clear();
315. if (organization_code > 1000){
316. organizations_file << "\n"; }
317. organizations_file << organization_code << " ";
318. organizations_file << organization_name << " | ";
319. organizations_file << publ_year << " | ";
320. organizations_file << organization_place << " | ";
321. organizations_file << organization_genre << " | ";
322. organizations_file << organization_platform << " | ";
323. organizations_file << organization_establisher << " | ";
324. organizations_file << organization_games << " | ";
325. organizations_file << organization_info << " | ";
326. cout << "Дані про організацію були успішно записані!" << endl;
327. organizations_file.close();

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						42
Зм.	Арк.	№ докум.	Підп.	Дата		

```

328. return;}
329. catch (const std::exception&) { cout << "Помилка! Не вдалося
    відкрити файл" << endl;}}
330. void Oread(int choise, string temp){
331. int i;
332. string temp2 = "";
333. cout << "Ви у функції читання даних про організації" << endl;
334. ifstream organizations_file(path);
335. try{
336. organizations_file.open(path);
337. Ragain:
338. i = 0;
339. organizations_file.clear();
340. organizations_file.seekg(0, ios_base::beg);
341. cout << "Введіть: \n1 - Щоб знайти організацію за
    ідентифікаційним кодом \n2 - Щоб вивести першу організацію в
    списку \nБудь-що інше - Щоб повернутись у головне меню" << endl;
342. cin >> temp;
343. choise = atoi(temp.c_str());
344. switch (choise){
345. case 1:
346. {i = 1;
347. cout << "Введіть ідентифікаційний код організації" << endl;
348. cin >> temp;
349. choise = atoi(temp.c_str());
350. while (!organizations_file.eof()){
351. if (organizations_file.eof()){
352. break;}
353. organizations_file >> temp;
354. if (atoi(temp.c_str()) == choise){
355. cout << "Організація із ідентифікаційним кодом: " <<
    atoi(temp.c_str()) << endl;
356. i = -1;
357. Next_org2:
358. cout << "\nНазва організації:\n";
359. choise = 0;
360. While_again:
361. temp = "";
362. while (temp != "|"){
363. organizations_file >> temp;
364. if (temp == "|"){
365. else{
366. cout << temp << " ";}}
367. choise++;
368. switch (choise){
369. case 1:{
370. cout << "\n";
371. break;}
372. case 2:{
373. cout << "року заснування\nМісце розташування:\n";
374. break;}
375. case 3:{

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						43
Зм.	Арк.	№ докум.	Підп.	Дата		

```

376. cout << "\nНайпопулярніші жанри ігор:\n";
377. break;}
378. case 4:{
379. cout << "\nНайпопулярніші платформи для створених ігор:\n";
380. break;}
381. case 5:{
382. cout << "\nЗасновник/ки організації:\n";
383. break;}
384. case 6:{
385. cout << "\nНайпопулярніші ігри організації:\n";
386. break;}
387. case 7:{
388. cout << "\nЗагальна інформація про організацію:\n";
389. break;}}
390. if (choise < 8){
391. goto While_again;}
392. if (i == 2){
393. cout << "\n\nВведіть: \n1 - Щоб повернутись у початок функції
      читання даних \n2 - Щоб вивести наступну організацію у
      списку\nБудь-що інше - Щоб повернутись у головне меню\n";
394. cin >> choise;
395. switch (choise){
396. case 1:{
397. goto Ragain;
398. break;}
399. case 2:{
400. organizations_file >> temp;
401. if (size(temp) != 4){
402. cout << "\n\n\nКінець списку\n\n\n";
403. return;}
404. cout << "Наступна організація у списку із ідентифікаційним
      кодом: " << temp << endl;
405. goto Next_org2;}}}}
406. if (i == -1){
407. cout << "\n\nВведіть: \n1 - Щоб повернутись у початок функції
      читання даних \nБудь-що інше - Щоб повернутись у головне меню\n";
408. cin >> choise;
409. if (choise == 1){
410. goto Ragain;}}}}
411. getline(organizations_file, temp);}
412. if (i == 1){
413. cout << "\nОрганізації із таким ідентифікаційним кодом неіснує"
      << endl;
414. goto Ragain;}
415. break;}
416. case 2:{
417. organizations_file.clear();
418. organizations_file.seekg(0, ios_base::beg);
419. organizations_file >> temp;
420. if (sizeof(temp) != 4){
421. cout << "\n\n\nОрганізації із таким ідентифікаційним кодом не
      існує\n\n\n\n";
422. return;}

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						44
Зм.	Арк.	№ докум.	Підп.	Дата		

```

423. cout << "Перша організація у списку із ідентифікаційним кодом: "
    << temp << endl;
424. choise = 0;
425. cout << "\nНазва організації:\n";
426. temp = "";
427. while (temp != "|"){
428. organizations_file >> temp;
429. if (temp == "|"){
430. else{
431. cout << temp << " ";}}
432. choise++;
433. cout << "\n";
434. i = 2;
435. goto While_again;}}
436. organizations_file.close();}
437. catch (const std::exception&){
438. cout << "Помилка! Не вдалося відкрити файл" << endl;}
439. return;}};

```

### class games

```

440. class games{
441. private:
442. int game_code = 0;
443. string game_publ_year, game_name, game_platform, game_genre,
    game_info, game_type, game_online, game_price, game_organization,
    gpath = "games_file.txt";
444. public:
445. void delete_games(int choise, string temp, int itemp, string
    str){
446. ifstream games_file(gpath, ios::binary);
447. Gwrong_code:
448. games_file.clear();
449. games_file.seekg(0, ios_base::beg);
450. cout << "Введіть ідентифікаційний код гри, інформацію про яку
    хочете видалити:" << endl;
451. cin >> temp;
452. choise = atoi(temp.c_str());
453. itemp = choise;
454. games_file.clear();
455. while (!games_file.eof()){
456. Gstart:
457. if (games_file.eof()){
458. cout << "\n\n\nГри із таким ідентифікаційним кодом
    неіснує\n\n\n";
459. goto Gwrong_code;}
460. games_file >> temp;
461. if (atoi(temp.c_str()) == choise){
462. cout << "\n\n\nГра із ідентифікаційним кодом: " <<
    atoi(temp.c_str()) << endl;
463. cout << "\nНазва гри:\n";
464. choise = 0;}
465. else{
466. getline(games_file, temp);
467. goto Gstart;}
468. while (temp != "|"){

```

					<b>КР.ІП-25.00.00.000 ПЗ</b>	Арк.
						45
Зм.	Арк.	№ докум.	Підп.	Дата		

```

469. temp = "";
470. games_file >> temp;
471. if (temp == "|"){
472. else{
473. cout << temp << " ";}}
474. break;}
475. cout << "\n\n\nВи впевненні, що хочете видалити інформацію про
    неї?\n\nВведіть:\n1 - Якщо ні\nБудь-що інше - Якщо так\n";
476. cin >> temp;
477. cin.ignore();
478. choise = atoi(temp.c_str());;
479. if (choise == 1){
480. goto Gwrong_code;}
481. choise = itemp;
482. games_file.clear();
483. games_file.seekg(0, ios_base::beg);
484. while (!games_file.eof()){
485. Gend:
486. games_file >> temp;
487. if (size(temp) != 3){
488. break;}
489. if (itemp == 1){
490. choise = atoi(temp.c_str()) - 1;
491. temp = to_string(choise);
492. str += temp;
493. getline(games_file, temp);
494. str += temp + "\n";
495. goto Gend;}
496. if (atoi(temp.c_str()) == choise){
497. getline(games_file, temp);
498. itemp = 1;}
499. else{
500. str += temp;
501. getline(games_file, temp);
502. str += temp + "\n";}}
503. ofstream Gout(gpath, ios::binary);
504. Gout << str;
505. cout << "\n\n\nЗміни успішно збережено!\n\n\n";
506. return;}
507. void edit_games(int choise, string temp, int itemp, string
    new_word, string str){
508. int i = 0, k = 1;
509. ifstream games_file(gpath, ios::binary);
510. Gwrong_code:
511. games_file.clear();
512. games_file.seekg(0, ios_base::beg);
513. cout << "Введіть ідентифікаційний код продукту, інформацію про
    який хочете змінити:" << endl;
514. cin >> temp;
515. choise = atoi(temp.c_str());;
516. itemp = choise;
517. games_file.clear();
518. while (!games_file.eof()){
519. start1:
520. if (games file.eof()){

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						46
Зм.	Арк.	№ докум.	Підп.	Дата		

```

521. cout << "\n\n\nГри із таким ідентифікаційним кодом
    неіснує\n\n\n";
522. goto Gwrong_code;
523. break;}
524. games_file >> temp;
525. if (atoi(temp.c_str()) == choise){
526. cout << "Гра із ідентифікаційним кодом: " << atoi(temp.c_str())
    << endl;
527. cout << "\nНазва гри:\n";
528. choise = 1;}
529. else{
530. getline(games_file, temp);
531. goto start1;}
532. Ggain:
533. temp = "";
534. while (temp != "|"){
535. games_file >> temp;
536. if (temp == "|"){
537. else{
538. cout << temp << " ";}}
539. choise++;
540. switch (choise){
541. case 2:{
542. cout << "\n";
543. break;}
544. case 3:{
545. cout << " Року створення\nЖанр гри:\n";
546. break;}
547. case 4:{
548. cout << "\nПлатформи, де гра доступна:\n";
549. break;}
550. case 5:{
551. cout << "\nТип гри:\n";
552. break;}
553. case 6:{
554. cout << "\nЧи є можливість онлайн гри:\n";
555. break;}
556. case 7:{
557. cout << "\nЦіна гри:\n";
558. break;}
559. case 8:{
560. cout << "грн\nНазва організації, яка створила продукт:\n";
561. break;}
562. case 9:{
563. cout << "\nЗагальна інформація про гру:\n";
564. break;}}
565. if (choise < 10){
566. goto Ggain;}
567. else{
568. break;}}
569. Gedit_again:

```

```

570. cout << "\n\nВведіть: \n1 - Щоб змінити назву гри \n2 - Щоб
    змінити рік створення гри \n3 - Щоб змінити жанр гри \n4 - Щоб
    змінити платформи, де гра доступна \n5 - Щоб змінити тип гри \n6
    - Щоб змінити можливість онлайн гри \n7 - Щоб змінити ціну гри
    \n8 - Щоб змінити назву організації, яка створила продукт\n9 -
    Щоб змінити загальну інформацію про гру \n10 - Щоб повернутись у
    меню" << endl;
571. cin >> temp;
572. choise = atoi(temp.c_str());
573. cin.ignore();
574. if (choise < 1){
575. cout << "\n\nНекоректний вибір, спробуйте знову";
576. goto Gedit_again;}
577. else
578. if (choise > 10){
579. cout << "\n\nНекоректний вибір, спробуйте знову";
580. goto Gedit_again;}
581. switch (choise){
582. case 1:{
583. cout << "Введіть нову назву гри:" << endl;
584. getline(cin, temp);
585. break;}
586. case 2:{
587. cout << "Введіть новий рік створення гри:" << endl;
588. getline(cin, temp);
589. break;}
590. case 3:{
591. cout << "Введіть новий жанр гри:" << endl;
592. getline(cin, temp);
593. break;}
594. case 4:{
595. cout << "Введіть нові платформи, де гра доступна:" << endl;
596. getline(cin, temp);
597. break;}
598. case 5:{
599. cout << "Введіть тип гри (2D, 3D ..):" << endl;
600. getline(cin, temp);
601. break;}
602. case 6:{
603. cout << "Введіть чи є можливість онлайн гри (*так* або *ні*):"
        << endl;
604. getline(cin, temp);
605. break;}
606. case 7:{
607. cout << "Введіть нову ціну гри (у гривнях):" << endl;
608. getline(cin, temp);
609. break;}
610. case 8:{
611. cout << "Введіть назву організації, яка створила продукт:" <<
        endl;
612. getline(cin, temp);
613. break;}
614. case 9:{
615. cout << "Введіть загальну інформацію про гру:" << endl;
616. getline(cin, temp);

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						48
Зм.	Арк.	№ докум.	Підп.	Дата		

```

617. break;}
618. case 10:{
619. return;}}
620. i = choise;
621. new_word = temp;
622. cout << "\n\nВи ввели:\n" << temp << "\n\n\n\n\nВведіть 1 - Щоб
    змінити 2 - Якщо усе введено правильно і щоб перезаписати у
    файл\n";
623. cin >> temp;
624. if (atoi(temp.c_str()) == 1){
625. goto Gedit_again;}
626. choise = itemp;
627. games_file.clear();
628. games_file.seekg(0, ios_base::beg);
629. while (!games_file.eof()){
630. games_file >> temp;
631. if (size(temp) != 3){
632. return;}
633. str += temp;
634. if (atoi(temp.c_str()) == choise){
635. for (k = 1; k < i; k++) {
636. getline(games_file, temp, '|');
637. str += temp + "|";}
638. if (k = i){
639. getline(games_file, temp, '|');
640. getline(games_file, temp);
641. str += " " + new_word + " |" + temp;}}
642. else {
643. getline(games_file, temp);
644. str += temp;
645. k = 1;}
646. str += "\n";}
647. ofstream Gout(gpath, ios::binary);
648. Gout << str;
649. cout << "\n\n\nЗміни успішно збережено!\n\n\n";
650. return;};
651. void Addgame(string temp, int choise){
652. cout << "Ви у функції додавання ігор в базу даних" << endl;
653. fstream games_file(gpath, ios::in | ios::out | ios::app);
654. try{
655. games_file.open(gpath);
656. Gagain:
657. games_file.clear();
658. cout << "Введіть назву ігри" << endl;
659. getline(cin, game_name);
660. cout << "Ви ввели: " << game_name << endl;
661. cout << "Введіть рік випуску" << endl;
662. cin >> temp;
663. game_publ_year = atoi(temp.c_str());
664. cin.ignore();
665. cout << "Ви ввели: " << game_publ_year << endl;
666. cout << "Введіть жанр продукту" << endl;
667. getline(cin, game_genre);
668. cout << "Ви ввели: " << game_genre << endl;

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						49
Зм.	Арк.	№ докум.	Підп.	Дата		



```

669. cout << "Введіть назви платформ, на яких продукт доступний" << endl;
670. getline(cin, game_platform);
671. cout << "Ви ввели: " << game_platform << endl;
672. cout << "Введіть тип гри (2D, 3D ..)" << endl;
673. getline(cin, game_type);
674. cout << "Ви ввели: " << game_type << endl;
675. cout << "Введіть *Так*, якщо у гри є можливість онлайн гри,
    *ні*, якщо немає" << endl;
676. getline(cin, game_online);
677. cout << "Ви ввели: " << game_online << endl;
678. cout << "Введіть ціну гри (у гривнях)" << endl;
679. getline(cin, game_price);
680. cout << "Ви ввели: " << game_price << endl;
681. cout << "Введіть назву організації, яка створила продукт" << endl;
682. getline(cin, game_organization);
683. cout << "Ви ввели: " << game_organization << endl;
684. cout << "Введіть загальну інформацію про ігру" << endl;
685. getline(cin, game_info);
686. cout << "Ви ввели: " << game_info << endl;
687. Gchange_again:
688. cout << "Назва ігри: " << game_name << "\nРік випуску продукту:
    " << game_publ_year << "\nЖанр продукту: " << game_genre <<
    "\nНазви платформ, на яких продукт доступний: " << game_platform
    << "\nТип гри: " << game_type << "\nМожливість гри онлайн: " <<
    game_online << "\nЦіна: " << game_price << " грн.\nОрганізація,
    яка створила продукт: " << game_organization << "\nЗагальна
    інформація: " << game_info << "\nЧи усе правильно введено і можна
    вносити дані у файл?\nВведіть: \n1 - Щоб змінити Назву продукту
    \n2 - Щоб змінити рік випуску \n3 - Щоб змінити жанр продукту
    \n4 - Щоб змінити Платформи, на яких достунпий продукт \n5 - Щоб
    змінити тип продукту \n6 - Щоб змінити можливість онлайн гри \n7
    - Щоб змінити ціну продукту \n8 - Щоб змінити організацію, яка
    створила продукт \n9 - Щоб змінити загальну інформацію про
    продукт \n10 - Щоб внести усі дані заново \n0 - Якщо усі дані
    введено вірно " << endl;
689. cin >> temp;
690. choise = atoi(temp.c_str());
691. switch (choise){
692. case 1:{
693. cout << "Введіть назву ігри" << endl;
694. cin.ignore();
695. getline(cin, game_name);
696. cout << "Ви ввели: " << game_name << endl;
697. break; }
698. case 2:{
699. cout << "Введіть рік випуску продукту" << endl;
700. cin.ignore();
701. getline(cin, game_publ_year);
702. cout << "Ви ввели: " << game_publ_year << endl;
703. break; }
704. case 3:{
705. cout << "Введіть жанр продукту" << endl;

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						50
Зм.	Арк.	№ докум.	Підп.	Дата		

```

706. cin.ignore();
707. getline(cin, game_genre);
708. cout << "Ви ввели: " << game_genre << endl;
709. break; }
710. case 4:{
711. cout << "Введіть назви платформ, на яких продукт доступний" <<
    endl;
712. cin.ignore();
713. getline(cin, game_platform);
714. cout << "Ви ввели: " << game_platform << endl;
715. break; }
716. case 5:{
717. cout << "Введіть тип продукту (2D, 3D ..)" << endl;
718. cin.ignore();
719. getline(cin, game_type);
720. cout << "Ви ввели: " << game_type << endl;
721. break; }
722. case 6:{
723. cout << "Введіть *Так*, якщо у гри є можливість онлайн гри,
    *ні*, якщо немає" << endl;
724. cin.ignore();
725. getline(cin, game_online);
726. cout << "Ви ввели: " << game_online << endl;
727. break; }
728. case 7:{
729. cout << "Введіть ціну гри (у гривнях)" << endl;
730. cin.ignore();
731. getline(cin, game_price);
732. cout << "Ви ввели: " << game_price << endl;
733. break; }
734. case 8:{
735. cout << "Введіть назву організації, яка створила продукт" <<
    endl;
736. cin.ignore();
737. getline(cin, game_organization);
738. cout << "Ви ввели: " << game_organization << endl;}
739. case 9:{
740. cout << "Введіть загальну інформацію про ігру" << endl;
741. cin.ignore();
742. getline(cin, game_info);
743. cout << "Ви ввели: " << game_info << endl;}
744. case 10: {
745. cin.ignore();
746. goto Gagain;}
747. case 0:{
748. break;}}
749. if ((choise < 10) && (choise > 0)){
750. cout << "Все правильно введено? Введіть: \n1 - якщо так \n2 -
    якщо ні або потрібно ще щось змінити" << endl;
751. cin >> choise;
752. if (choise == 2) {
753. goto Gchange_again;}}

```

```

754. temp = "";
755. games_file.clear();
756. string temp1;
757. while (!games_file.eof()){
758. games_file >> temp;
759. getline(games_file, temp1); }
760. if (temp == ""){
761. game_code = 100;}
762. else{
763. game_code = atoi(temp.c_str());
764. game_code++;}
765. games_file.clear();
766. if (game_code > 100){
767. games_file << "\n";}
768. games_file << game_code << " ";
769. games_file << game_name << " | ";
770. games_file << game_publ_year << " | ";
771. games_file << game_genre << " | ";
772. games_file << game_platform << " | ";
773. games_file << game_type << " | ";
774. games_file << game_online << " | ";
775. games_file << game_price << " | ";
776. games_file << game_organization << " | ";
777. games_file << game_info << " | ";
778. cout << "Ігру успішно записано в базу даних!" << endl; }
779. catch (const std::exception&){
780. cout << "Помилка! Не вдалося відкрити файл" << endl; }
781. return;};
782. void Gread(int choise, string temp){
783. int i;
784. string temp2 = "";
785. cout << "Ви у функції читання даних про ігри" << endl;
786. ifstream games_file(gpath);
787. try{
788. games_file.open(gpath);
789. RGagain:
790. games_file.clear();
791. games_file.seekg(0, ios_base::beg);
792. i = 0;
793. cout << "Введіть: \n1 - Щоб знайти гру за ідентифікаційним кодом
       \n2 - Щоб вивести першу гру в списку \nБудь-що інше - Щоб
       повернутись у головне меню" << endl;
794. cin >> temp;
795. choise = atoi(temp.c_str());
796. switch (choise){
797. case 1:{
798. i = 1;
799. cout << "Введіть ідентифікаційний код гри" << endl;
800. cin >> temp;
801. choise = atoi(temp.c_str());
802. while (!games_file.eof()){
803. if (games_file.eof()){
804. break;}
805. games_file >> temp;

```

```

806. if (atoi(temp.c_str()) == choise){
807. cout << "Гра із ідентифікаційним кодом: " << atoi(temp.c_str())
    << endl;
808. i = -1;
809. Next_game:
810. cout << "\nНазва гри:\n";
811. choise = 0;
812. While_again_game:
813. temp = "";
814. while (temp != "|"){
815. games_file >> temp;
816. if (temp == "|"){
817. else{
818. cout << temp << " ";}}
819. choise++;
820. switch (choise){
821. case 1:{
822. cout << "\n";
823. break;}
824. case 2:{
825. cout << "року створення\nЖанр гри:\n";
826. break;}
827. case 3:{
828. cout << "\nПлатформи, де ігра доступна:\n";
829. break;}
830. case 4:{
831. cout << "\nТип гри:\n";
832. break;}
833. case 5:{
834. cout << "\nЧи є можливісь онлайн гри:\n";
835. break;}
836. case 6:{
837. cout << "\nЦіна гри:\n";
838. break;}
839. case 7:{
840. cout << "грн.\nНазва організації, яка створила продукт:\n";
841. break;}
842. case 8:{
843. cout << "\nЗагальна інформація про гру:\n";
844. break;}}
845. if (choise < 9){
846. goto While_again_game;}
847. if (i == 2){
848. cout << "\n\nВведіть: \n1 - Щоб повернутись у початок функції
    читання даних \n2 - Щоб вивести наступну організацію у
    списку\nБудь-що інше - Щоб повернутись у головне меню\n";
849. cin >> choise;
850. switch (choise){
851. case 1:{
852. goto RGagain;
853. break;}
854. case 2:{
855. games_file >> temp;
856. if (size(temp) != 3){

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						53
Зм.	Арк.	№ докум.	Підп.	Дата		

```

857. cout << "\n\nКінець списку\n\n\n";
858. return;
859. cout << "Наступна гра у списку із ідентифікаційним кодом: " <<
    temp << endl;
860. goto Next_game;}}
861. if (i == -1){
862. cout << "\n\nВведіть: \n1 - Щоб повернутись у початок функції
    читання даних \nБудь-що інше - Щоб повернутись у головне меню\n";
863. cin >> choise;
864. if (choise == 1){
865. goto RGagain;}}
866. getline(games_file, temp);
867. if (i == 1){
868. cout << "\nГри із таким ідентифікаційним кодом неіснує" << endl;
869. goto RGagain;
870. break;}
871. case 2:{
872. games_file.clear();
873. games_file.seekg(0, ios_base::beg);
874. games_file >> temp;
875. if (sizeof(temp) != 3){
876. cout << "\n\nГри із таким ідентифікаційним кодом не
    існує\n\n\n\n";
877. return;}
878. cout << "Перша гра у списку із ідентифікаційним кодом: " << temp
    << endl;
879. choise = 0;
880. cout << "\nНазва гри:\n";
881. temp = "";
882. while (temp != "|"){
883. games_file >> temp;
884. if (temp == "|"){
885. else{
886. cout << temp << " ";}}
887. choise++;
888. cout << "\n";
889. i = 2;
890. goto While_again_game;}}
891. games_file.close();
892. catch (const std::exception&){
893. cout << "Помилка! Не вдалося відкрити файл" << endl;}
894. return;}};

```

### class administrators

```

895. class administrators{
896. private:
897. int admin_code = 0;
898. string date_start, date_finish, admin_type, admin_name,
    admin_surname, admin_middlename, admin_phone, admin_settlement,
    admin_house, admin_flat, admin_street, admin_post, admin_email,
    admin_country, apath = "Administrators_file.txt";

```

					<b>КР.ІП-25.00.00.000 ПЗ</b>	Арк.
						54
Зм.	Арк.	№ докум.	Підп.	Дата		

```

899. public:
900. void delete_administrator(int choise, string temp, int itemp,
    string str){
901. ifstream administrators_file(apath, ios::binary);
902. Awrong_code:
903. administrators_file.clear();
904. administrators_file.seekg(0, ios_base::beg);
905. cout << "Введіть ідентифікаційний код адміністратора, інформацію
    про якого хочете видалити:" << endl;
906. cin >> temp;
907. choise = atoi(temp.c_str());;
908. itemp = choise;
909. administrators_file.clear();
910. while (!administrators_file.eof()){
911. start1:
912. if (administrators_file.eof()){
913. cout << "\n\nАдміністратора із таким ідентифікаційним кодом
    не існує\n\n";
914. goto Awrong_code;}
915. administrators_file >> temp;
916. if (atoi(temp.c_str()) == choise){
917. cout << "\n\nАдміністратор із ідентифікаційним кодом: " <<
    atoi(temp.c_str()) << endl;
918. cout << "\nПовне ім'я адміністратора:\n";
919. choise = 0;}
920. else{
921. getline(administrators_file, temp);
922. goto start1;}
923. for (int t = 0; t < 3; t++){
924. temp = "";
925. while (temp != "|"){
926. administrators_file >> temp;
927. if (temp == "|"){}}
928. else{
929. cout << temp << " ";}}}
930. break;}
931. cout << "\n\nВи впевненні, що хочете видалити інформацію про
    нього?\n\nВведіть:\n1 - Якщо ні\nБудь-що інше - Якщо так\n";
932. cin >> temp;
933. cin.ignore();
934. choise = atoi(temp.c_str());;
935. if (choise == 1){
936. goto Awrong_code;}
937. choise = itemp;
938. itemp = 0;
939. administrators_file.clear();
940. administrators_file.seekg(0, ios_base::beg);
941. while (!administrators_file.eof()){
942. Aend:
943. administrators_file >> temp;
944. if (size(temp) != 2){
945. break;}
946. if (itemp == 1){
947. choise = atoi(temp.c_str()) - 1;

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						55
Зм.	Арк.	№ докум.	Підп.	Дата		

```

948. temp = to_string(choise);
949. str += temp;
950. getline(administrators_file, temp);
951. str += temp + "\n";
952. goto Aend;}
953. if (atoi(temp.c_str()) == choise){
954. getline(administrators_file, temp);
955. itemp = 1;}
956. else{
957. str += temp;
958. getline(administrators_file, temp);
959. str += temp + "\n";}}
960. ofstream aut(apath, ios::binary);
961. aut << str;
962. cout << "\n\nЗміни успішно збережено!\n\n";
963. return ;}
964. void edit_administrator(int choise, string temp, int itemp,
    string new_word, string str){
965. int i = 0, k = 1;
966. ifstream administrators_file(apath, ios::binary);
967. wrong_code:
968. administrators_file.clear();
969. administrators_file.seekg(0, ios_base::beg);
970. cout << "Введіть ідентифікаційний код адміністратора, інформацію
    про якого хочете змінити:" << endl;
971. cin >> temp;
972. choise = atoi(temp.c_str());;
973. itemp = choise;
974. administrators_file.clear();
975. while (!administrators_file.eof()){
976. start1:
977. if (administrators_file.eof()){
978. cout << "\n\nАдміністратора із таким ідентифікаційним кодом
    неіснує\n\n";
979. goto wrong_code;
980. break;}
981. administrators_file >> temp;
982. if (atoi(temp.c_str()) == choise){
983. cout << "Адміністратор із ідентифікаційним кодом: " <<
    atoi(temp.c_str()) << endl;
984. cout << "\nПовне ім'я адміністратора:\n";
985. choise = 0;}
986. else{
987. getline(administrators_file, temp);
988. goto start1;}
989. again:
990. temp = "";
991. while (temp != "|"){
992. administrators_file >> temp;
993. if (temp == "|"){}}
994. else{
995. cout << temp << " ";}}
996. choise++;

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						56
Зм.	Арк.	№ докум.	Підп.	Дата		

```

997. switch (choise){
998. case 3:{
999. cout << "\nТип роботи, за який відповідальний:\n";
1000.break;}
1001.case 4:{
1002.cout << "\nПовний номер телефону:\n";
1003.break;}
1004.case 5:{
1005.cout << "\nЕлектронна адреса:\n";
1006.break;}
1007.case 6:{
1008.cout << "\nКраїна, у якій проживає:\n";
1009.break;}
1010.case 7:{
1011.cout << "\nМісто/село, у якому проживає:\n";
1012.break;}
1013.case 8:{
1014.cout << "\nНомер будинку:\n";
1015.break;}
1016.case 9:{
1017.cout << "\nНомер квартири:\n";
1018.break;}
1019.case 10:{
1020.cout << "\nВулиця:\n";
1021.break;}
1022.case 11:{
1023.cout << "\nПоштовий індекс:\n";
1024.break;}
1025.case 12:{
1026.cout << "\nДата прийняття на роботу:\n";
1027.break;}
1028.case 13:{
1029.cout << "\nДата звільнення з роботи:\n";
1030.break;}}}
1031.if (choise < 14){
1032.goto egain;}
1033.else{
1034.break;}}
1035.Aedit_again:
1036.cout << "\n\nВведіть: \n1 - Щоб змінити ім'я адміністратора
\n2 - Щоб змінити прізвище адміністратора \n3 - Щоб змінити по-
батькові адміністратора \n4 - Щоб змінити тип роботи, за який
відповідальний \n5 - Щоб змінити повний номер телефону \n6 - Щоб
змінити електронну адресу \n7 - Щоб змінити країну, у якій
проживає \n8 - Щоб змінити місто/село у якому проживає \n9 - Щоб
змінити номер будинку, у якому проживає \n10 - Щоб змінити номер
квартири \n11 - Щоб змінити вулицю \n12 - Щоб змінити поштовий
індекс \n13 - Щоб змінити дату прийняття на роботу \n14 - Щоб
змінити дату звільнення з роботи" << endl;
1037.cin >> temp;
1038.cin.ignore();
1039.choise = atoi(temp.c_str());;
1040.if (choise < 1){

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						57
Зм.	Арк.	№ докум.	Підп.	Дата		



```

1041.cout << "\n\n\nНекоректний вибір, спробуйте знову";
1042.goto Aedit_again;}
1043.else
1044.if (choise > 14){
1045.cout << "\n\n\nНекоректний вибір, спробуйте знову";
1046.goto Aedit_again;}
1047.switch (choise){
1048.case 1:{
1049.cout << "Введіть нове ім'я адміністратора:" << endl;
1050.getline(cin, temp);
1051.break;}
1052.case 2:{
1053.cout << "Введіть нове прізвище адміністратора:" << endl;
1054.getline(cin, temp);
1055.break;}
1056.case 3:{
1057.cout << "Введіть нове по-батькові адміністратора:" << endl;
1058.getline(cin, temp);
1059.break;}
1060.case 4:{
1061.cout << "Введіть новий тип роботи, за який відповідальний:" <<
    endl;
1062.getline(cin, temp);
1063.break;}
1064.case 5:{
1065.cout << "Введіть новий повний номер телефону:" << endl;
1066.getline(cin, temp);
1067.break;}
1068.case 6:{
1069.cout << "Введіть нову електронну адресу:" << endl;
1070.getline(cin, temp);
1071.break;}
1072.case 7:{
1073.cout << "Введіть нову країну проживання:" << endl;
1074.getline(cin, temp);
1075.break;}
1076.case 8:{
1077.cout << "Введіть нове місто/село, у якому проживає:" << endl;
1078.getline(cin, temp);
1079.break;}
1080.case 9:{
1081.cout << "Введіть новий номер будинку:" << endl;
1082.getline(cin, temp);
1083.break;}
1084.case 10:{
1085.cout << "Введіть новий номер квартири (введіть *1*, якщо
    проживає у власному будинку" << endl;
1086.cin >> temp;
1087.cin.ignore();
1088.if (temp == "1"){
1089.temp = "Проживає у власному будинку";}
1090.cout << "Введіть новий номер квартири:" << endl;
1091.break;}
1092.case 11:{

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						58
Зм.	Арк.	№ докум.	Підп.	Дата		

```

1093.cout << "Введіть нову вулицю:" << endl;
1094.getline(cin, temp);
1095.break;}
1096.case 12:{
1097.cout << "Введіть новий поштовий індекс:" << endl;
1098.getline(cin, temp);
1099.break;}
1100.case 13:{
1101.cout << "Введіть нову дату прийняття на роботу:" << endl;
1102.getline(cin, temp);
1103.break;}
1104.case 14:{
1105.cout << "Введіть нову дату звільнення з роботи (введіть *1*,
        якщо він досі працює на своїй посаді)" << endl;
1106.cin >> temp;
1107.cin.ignore();
1108.if (temp == "1"){
1109.temp = "Досі працює на своїй посаді";}
1110.break;}}
1111.i = choise;
1112.new_word = temp;
1113.cout << "\n\nВи ввели:\n" << temp << "\n\n\n\nВведіть 1 - Щоб
        змінити 2 - Якщо усе введено правильно і щоб перезаписати у
        файл\n";
1114.cin >> temp;
1115.if (atoi(temp.c_str()) == 1){
1116.goto Aedit_again;}
1117.choise = itemp;
1118.administrators_file.clear();
1119.administrators_file.seekg(0, ios_base::beg);
1120.while (!administrators_file.eof()){
1121.administrators_file >> temp;
1122.if (size(temp) != 2){
1123.return;}
1124.str += temp;
1125.if (atoi(temp.c_str()) == choise){
1126.for (k = 1; k < i; k++) {
1127.getline(administrators_file, temp, '|');
1128.str += temp + "|";}
1129.if (k = i){
1130.getline(administrators_file, temp, '|');
1131.getline(administrators_file, temp);
1132.str += " " + new_word + " |" + temp;}}
1133.else {
1134.getline(administrators_file, temp);
1135.str += temp;
1136.k = 1;}
1137.str += "\n";}
1138 ofstream out(aphath, ios::binary);
1139.out << str;
1140.cout << "\n\n\nЗміни успішно збережено!\n\n\n";
1141.return;};
1142.void add_administrator(int choise, string temp) {

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						59
Зм.	Арк.	№ докум.	Підп.	Дата		

```

1143.cout << "Ви у функції додавання інформації про адміністраторів
у базу даних" << endl;
1144.fstream administrators_file(apath, ios::in | ios::out |
ios::app);
1145.try{
1146.administrators_file.open(apath);
1147.Aallagain:
1148.administrators_file.clear();
1149.cout << "Введіть ім'я адміністратора" << endl;
1150.getline(cin, admin_name);
1151.cout << "Ви ввели: " << admin_name << endl;
1152.cout << "Введіть прізвище адміністратора" << endl;
1153.getline(cin, admin_surname);
1154.cout << "Ви ввели: " << admin_surname << endl;
1155.cout << "Введіть по-батькові адміністратора" << endl;
1156.getline(cin, admin_middlename);
1157.cout << "Ви ввели: " << admin_middlename << endl;
1158.cout << "Введіть за який тип роботи він відповідальний" << endl;
1159.getline(cin, admin_type);
1160.cout << "Ви ввели: " << admin_type << endl;
1161.cout << "Введіть його повний номер телефону" << endl;
1162.getline(cin, admin_phone);
1163.cout << "Ви ввели: " << admin_phone << endl;
1164.cout << "Введіть його електронну адресу" << endl;
1165.getline(cin, admin_email);
1166.cout << "Ви ввели: " << admin_email << endl;
1167.cout << "Введіть країну, у якій він зараз проживає" << endl;
1168.getline(cin, admin_country);
1169.cout << "Ви ввели: " << admin_country << endl;
1170.cout << "Введіть його місце проживання(місто/село)" << endl;
1171.getline(cin, admin_settlement);
1172.cout << "Ви ввели: " << admin_settlement << endl;
1173.cout << "Введіть номер його будинку" << endl;
1174.getline(cin, admin_house);
1175.cout << "Ви ввели: " << admin_house << endl;
1176.cout << "Введіть номер його квартири (введіть *1*, якщо проживає
у власному будинку" << endl;
1177.cin >> admin_flat;
1178.cin.ignore();
1179.if (admin_flat == "1") {
1180.admin_flat = "Проживає у власному будинку";}
1181.cout << "Ви ввели: " << admin_flat << endl;
1182.cout << "Введіть його вулицю" << endl;
1183.getline(cin, admin_street);
1184.cout << "Ви ввели: " << admin_street << endl;
1185.cout << "Введіть його поштовий код" << endl;
1186.getline(cin, admin_post);
1187.cout << "Ви ввели: " << admin_post << endl;
1188.cout << "Введіть дату прийняття на роботу (наприклад:
01.01.2020)" << endl;
1189.getline(cin, date_start);
1190.cout << "Ви ввели: " << date_start << endl;
1191.cout << "Введіть дату звільнення з роботи (введіть *1*, якщо він
досі працює на своїй посаді)" << endl;

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						60
Зм.	Арк.	№ докум.	Підп.	Дата		

```

1192.cin >> date_finish;
1193.cin.ignore();
1194.if (date_finish == "1"){
1195.date_finish = "Досі працює на своїй посаді";}
1196.cout << "Ви ввели: " << date_finish << endl;
1197.Aagain:
1198.cout << "Ім'я адміністратора: " << admin_name << "\nПрізвище
адміністратора: " << admin_surname << "\nПо-батькові
адміністратора: " << admin_middlename << "\nТип роботи, за який
відповідальний: " << admin_type << "\nПовний номер телефону: " <<
admin_phone << "\nЕлектронна адреса: " << admin_email <<
"\nКраїна, у якій проживає: " << admin_country << "\nМісто/село, я
у якому проживає: " << admin_settlement << "\nНомер будинку, у
якому проживає: " << admin_house << "\nНомер квартири: " <<
admin_flat << "\nВулиця: " << admin_street << "\nПоштовий індекс "
<< admin_post << "\nДата прийняття на роботу: " << date_start <<
"\nДата звільнення з роботи: " << date_finish << "\nЧи усе
правильно введено і можна вносити дані у файл?\nВведіть: \n1 -
Щоб змінити ім'я адміністратора \n2 - Щоб змінити прізвище
адміністратора \n3 - Щоб змінити по-батькові адміністратора \n4
- Щоб змінити тип роботи, за який відповідальний \n5 - Щоб
змінити повний номер телефону \n6 - Щоб змінити електронну адресу
\n7 - Щоб змінити країну, у якій проживає \n8 - Щоб змінити
місто/село у якому проживає \n9 - Щоб змінити номер будинку, у
якому проживає \n10 - Щоб змінити номер квартири \n11 - Щоб
змінити вулицю \n12 - Щоб змінити поштовий індекс \n13 - Щоб
змінити дату прийняття на роботу \n14 - Щоб змінити дату
звільнення з роботи \n15 - Щоб внести усі дані заново \n0 - Якщо
усі дані введено вірно " << endl;
1199.cin >> temp;
1200.choise = atoi(temp.c_str());
1201.switch (choise){
1202.case 1:{
1203.cout << "Введіть ім'я адміністратора" << endl;
1204.cin.ignore();
1205.getline(cin, admin_name);
1206.cout << "Ви ввели: " << admin_name << endl;
1207.break;}
1208.case 2:
1209.{cout << "Введіть прізвище адміністратора" << endl;
1210.cin.ignore();
1211.getline(cin, admin_surname);
1212.cout << "Ви ввели: " << admin_surname << endl;
1213.break; }
1214.case 3:{
1215.cout << "Введіть по-батькові адміністратора" << endl;
1216.cin.ignore();
1217.getline(cin, admin_middlename);
1218.cout << "Ви ввели: " << admin_middlename << endl;
1219.break; }
1220.case 4:{
1221.cout << "Введіть тип роботи, за який відповідальний" << endl;
1222.cin.ignore();
1223.getline(cin, admin_type);

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						61
Зм.	Арк.	№ докум.	Підп.	Дата		

```

1224.cout << "Ви ввели: " << admin_type << endl;
1225.break; }
1226.case 5:{
1227.cout << "Введіть повний номер телефону" << endl;
1228.cin.ignore();
1229.getline(cin, admin_phone);
1230.cout << "Ви ввели: " << admin_phone << endl;
1231.break; }
1232.case 6:{
1233.cout << "Введіть електронну адресу" << endl;
1234.cin.ignore();
1235.getline(cin, admin_email);
1236.cout << "Ви ввели: " << admin_email << endl;
1237.break; }
1238.case 7:{
1239.cout << "Введіть країну, у якій він проживає" << endl;
1240.cin.ignore();
1241.getline(cin, admin_country);
1242.cout << "Ви ввели: " << admin_country << endl;
1243.break; }
1244.case 8:{
1245.cout << "Введіть місто/село, у якому він проживає" << endl;
1246.cin.ignore();
1247.cin >> admin_settlement;
1248.cout << "Ви ввели: " << admin_settlement << endl;}}
1249.case 9:{
1250.cout << "Введіть номер будинку, у якому він проживає" << endl;
1251.cin.ignore();
1252.cin >> admin_house;
1253.cout << "Ви ввели: " << admin_house << endl;}}
1254.case 10:{
1255.cout << "Введіть номер його квартири (введіть *1*, якщо проживає
    у власному будинку" << endl;
1256.cin >> admin_flat;
1257.cin.ignore();
1258.if (admin_flat == "1"){
1259.admin_flat = "Проживає у власному будинку";}
1260.cout << "Ви ввели: " << admin_flat << endl;
1261.cin >> admin_flat;
1262.cout << "Ви ввели: " << admin_flat << endl;}}
1263.case 11:{
1264.cout << "Введіть вулицю, на якій він проживає" << endl;
1265.cin.ignore();
1266.cin >> admin_street;
1267.cout << "Ви ввели: " << admin_street << endl;}}
1268.case 12:{
1269.cout << "Введіть його поштовий індекс" << endl;
1270.cin.ignore();
1271.cin >> admin_post;
1272.cout << "Ви ввели: " << admin_post << endl;}}
1273.case 13:{
1274.cout << "Введіть дату прийняття на роботу" << endl;

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						62
Зм.	Арк.	№ докум.	Підп.	Дата		

```

1275.cin.ignore();
1276.cin >> date_start;
1277.cout << "Ви ввели: " << date_start << endl;
1278.case 14:{
1279.cout << "Введіть дату звільнення з роботи (введіть *1*, якщо він
досі працює на своїй посаді)" << endl;
1280.cin >> date_finish;
1281.cin.ignore();
1282.if (date_finish == "1"){
1283.date_finish = "Досі працює на своїй посаді";}
1284.cout << "Ви ввели: " << date_finish << endl;
1285.case 15:{
1286.goto Aallagain;}
1287.case 0:
1288.{break;}}
1289.if ((choise < 15) && (choise > 0)){
1290.cout << "Все правильно введено? Введіть: \n1 - якщо так \n2 -
якщо ні або потрібно ще щось змінити" << endl;
1291.cin >> choise;
1292.if (choise == 2) {
1293.goto Aagain;}}
1294.temp = "";
1295.administrators_file.clear();
1296.string temp1;
1297.while (!administrators_file.eof()){
1298.administrators_file >> temp;
1299.getline(administrators_file, temp1);}
1300.if (temp == ""){
1301.admin_code = 10;}
1302.else{
1303.admin_code = atoi(temp.c_str());
1304.admin_code++;}
1305.administrators_file.clear();
1306.if (admin_code > 10){
1307.administrators_file << "\n";}
1308.administrators_file << admin_code << " ";
1309.administrators_file << admin_name << " | ";
1310.administrators_file << admin_surname << " | ";
1311.administrators_file << admin_middlename << " | ";
1312.administrators_file << admin_type << " | ";
1313.administrators_file << admin_phone << " | ";
1314.administrators_file << admin_email << " | ";
1315.administrators_file << admin_country << " | ";
1316.administrators_file << admin_settlement << " | ";
1317.administrators_file << admin_house << " | ";
1318.administrators_file << admin_flat << " | ";
1319.administrators_file << admin_street << " | ";
1320.administrators_file << admin_post << " | ";
1321.administrators_file << date_start << " | ";
1322.administrators_file << date_finish << " | ";
1323.cout << "Дані про адміністратора були успішно записані!" <<
endl;
1324.administrators_file.close();

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						63
Зм.	Арк.	№ докум.	Підп.	Дата		

```

1325.return;}
1326.catch (const std::exception&){
1327.cout << "Помилка! Не вдалося відкрити файл" << endl;}}
1328.void Aread(int choise, string temp){
1329.int i;
1330.string temp2 = "";
1331.cout << "Ви у функції читання даних про Адміністраторів" <<
    endl;
1332 ifstream administrators_file(apath);
1333.try{
1334.administrators_file.open(apath);
1335.Aagain:
1336.i = 0;
1337.administrators_file.clear();
1338.administrators_file.seekg(0, ios_base::beg);
1339.cout << "Введіть: \n1 - Щоб знайти Адміністратора за
    ідентифікаційним кодом \n2 - Щоб вивести першого Адміністратора в
    списку \nБудь-що інше - Щоб повернутись у головне меню" << endl;
1340.cin >> temp;
1341.choise = atoi(temp.c_str());
1342.switch (choise){
1343.case 1:{
1344.i = 1;
1345.cout << "Введіть ідентифікаційний код Адміністратора" << endl;
1346.cin >> temp;
1347.choise = atoi(temp.c_str());
1348.while (!administrators_file.eof()){
1349.if (administrators_file.eof()){
1350.break;}
1351.administrators_file >> temp;
1352.if (atoi(temp.c_str()) == choise){
1353.cout << "Адміністратор із ідентифікаційним кодом: " <<
    atoi(temp.c_str()) << endl;
1354.i = -1;
1355.Next_adm2:
1356.cout << "\nПовне ім'я адміністратора:\n";
1357.choise = 0;
1358.While_Aagain:
1359.temp = "";
1360.while (temp != "|"){
1361.administrators_file >> temp;
1362.if (temp == "|"){}}
1363.else{
1364.cout << temp << " ";}}
1365.choise++;
1366.switch (choise){
1367.case 3:{
1368.cout << "\nТип роботи, за який відповідальний:\n";
1369.break;}
1370.case 4:{
1371.cout << "\nПовний номер телефону:\n";
1372.break;}
1373.case 5:{

```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						64
Зм.	Арк.	№ докум.	Підп.	Дата		

```

1374.cout << "\nЕлектронна адреса:\n";
1375.break;}
1376.case 6:{
1377.cout << "\nКраїна, у якій проживає:\n";
1378.break;}
1379.case 7:{
1380.cout << "\nМісто/село, у якому проживає:\n";
1381.break;}
1382.case 8:{
1383.cout << "\nНомер будинку:\n";
1384.break;}
1385.case 9:{
1386.cout << "\nНомер квартири:\n";
1387.break;}
1388.case 10:{
1389.cout << "\nВулиця:\n";
1390.break;}
1391.case 11:{
1392.cout << "\nПоштовий індекс:\n";
1393.break;}
1394.case 12:{
1395.cout << "\nДата прийняття на роботу:\n";
1396.break;}
1397.case 13:{
1398.cout << "\nДата звільнення з роботи:\n";
1399.break;}}
1400.if (choise < 14){
1401.goto While_Aagain;}
1402.if (i == 2){
1403.cout << "\n\nВведіть: \n1 - Щоб повернутись у початок функції
        читання даних \n2 - Щоб вивести інформацію про наступного
        адміністратора у списку\nБудь-що інше - Щоб повернутись у головне
        меню\n";
1404.cin >> choise;
1405.switch (choise){
1406.case 1:{
1407.goto Aagain;
1408.break;}
1409.case 2:{
1410.administrators_file >> temp;
1411.if (size(temp) != 2){
1412.cout << "\n\n\nКінець списку\n\n\n";
1413.return;}
1414.cout << "Наступна організація у списку із ідентифікаційним
        кодом: " << temp << endl;
1415.goto Next_adm2;}}
1416.if (i == -1){
1417.cout << "\n\nВведіть: \n1 - Щоб повернутись у початок функції
        читання даних \nБудь-що інше - Щоб повернутись у головне меню\n";
1418.cin >> choise;
1419.if (choise == 1){
1420.goto Aagain;}}
1421.getline(administrators_file, temp);}
1422.if (i == 1){

```



```

1423.cout << "\nАдміністратора із таким ідентифікаційним кодом
      неіснує" << endl;
1424.goto Aagain;}
1425.break;}
1426.case 2:{
1427.administrators_file.clear();
1428.administrators_file.seekg(0, ios_base::beg);
1429.administrators_file >> temp;
1430.if (sizeof(temp) != 2) {
1431.cout << "\n\n\nАдміністратора із таким ідентифікаційним кодом не
      існує\n\n\n\n";
1432.return;}
1433.cout << "Перший Адміністратор у списку із ідентифікаційним
      кодом: " << temp << endl;
1434.choise = 0;
1435.cout << "\nПовне ім'я Адміністратора:\n";
1436.temp = "";
1437.while (temp != "|"){
1438.administrators_file >> temp;
1439.if (temp == "|"){}
1440.else{
1441.cout << temp << " ";}}
1442.choise++;
1443.i = 2;
1444.goto While_Aagain;}}
1445.administrators_file.close();}
1446.catch (const std::exception&){
1447.cout << "Помилка! Не вдалося відкрити файл" << endl;}
1448.return;}};
1449.void add_menu(int choise, string temp){
1450.while (true) {
1451.cout << "\nВи у меню додання, введіть потрібну цифру, щоб
      обрати подальшу функцію: \n1 - База даних організацій \n2 - База
      даних продуктів \n3 - База даних адміністраторів \n4 - Повернутись
      у головне меню\n";
1452.cin >> choise;
1453.switch (choise){
1454.case 1:{
1455.cin.ignore();
1456.organizations org;
1457.org.AddOrganization(choise, temp);
1458.break;}
1459.case 2:{
1460.cin.ignore();
1461.games gam;
1462.gam.Addgame(temp, choise);
1463.break;}
1464.case 3:{
1465.cin.ignore();
1466.administrators adm;
1467.adm.add_administrator(choise, temp);
1468.break;}
1469.case 4:{
1470.cout << endl;

```

```

1471.return;}};}}
1472.void read_menu(int choise, string temp){
1473.while (true) {
1474.cout << "Ви у меню читання, введіть потрібну цифру, щоб обрати
    подальшу функцію: \nБаза даних: \n1 - Організацій \n2 - Продуктів
    \n3 - Адміністраторів \n4 - Повернутись у головне меню" << endl;
1475.cin >> choise;
1476.switch (choise){
1477.case 1:{
1478.organizations org;
1479.org.Oread(choise, temp);
1480.break;}
1481.case 2:{
1482.games gam;
1483.gam.Gread(choise, temp);
1484.break;}
1485.case 3:{
1486.administrators adm;
1487.adm.Aread(choise, temp);
1488.break;}
1489.case 4:{
1490.return;}};}}
1491.void edit_menu(string temp, int choise){
1492.int itemp = 0;
1493.string new_word, str;
1494.while (true){
1495.new_word = "";
1496.str = "";
1497.cout << "Ви у меню редагування бази даних, введіть потрібну
    цифру, щоб обрати подальшу функцію: \n1 - Редагувати базу даних
    організацій \n2 - Редагувати базу даних ігор \n3 - Редагувати базу
    даних адміністраторів \n4 - повернутись у головне меню" << endl;
1498.cin >> temp;
1499.choise = atoi(temp.c_str());
1500.cin.ignore();
1501.switch (choise){
1502.case 1:{
1503.organizations org;
1504.org.edit_organizations(choise, temp, itemp, new_word, str);
1505.break;}
1506.case 2:{
1507.games gam;
1508.gam.edit_games(choise, temp, itemp, new_word, str);
1509.break;}
1510.case 3:{
1511.administrators adm;
1512.adm.edit_administrator(choise, temp, itemp, new_word, str);}
1513.case 4:{
1514.return;}};}}
1515.void delete_menu(int choise, string temp){
1516.int itemp = 0;
1517.string str;
1518.while (true){
1519.str = "";

```

```

1520.cout << "Ви у меню видалення інформації з бази даних, введіть
    потрібну цифру, щоб обрати подальшу функцію: \n1 - Базу даних
    організацій \n2 - База даних ігор \n3 - База даних адміністраторів
    \n4 - повернутись у головне меню" << endl;
1521.cin >> temp;
1522.choise = atoi(temp.c_str());
1523.cin.ignore();
1524.switch (choise){
1525.case 1:{
1526.organizations org;
1527.org.delete_organization(choise, temp, itemp, str);
1528.break;}
1529.case 2:{
1530.games gam;
1531.gam.delete_games(choise, temp, itemp, str);
1532.break;}
1533.case 3:{
1534.administrators adm;
1535.adm.delete_administrator(choise, temp, itemp, str);
1536.break;}
1537.case 4:{
1538.return;}}}}
1539.void main_menu(int choise, string temp){
1540.while (true) {
1541.cout << "Ви у головному меню, введіть потрібну цифру, щоб обрати
    подальшу функцію: \n1 - Додати дані в базу даних \n2 - Переглянути
    вміст або інформацію \n3 - Редагувати бази даних \n4 - Видалити
    інформацію з бази даних\n5 - Вийти з головного меню " << endl;
1542.cin >> temp;
1543.choise = atoi(temp.c_str());
1544.switch (choise){
1545.case 1:{
1546.cin.ignore();
1547.add_menu(choise, temp);
1548.break;}
1549.case 2:{
1550.cin.ignore();
1551.read_menu(choise, temp);
1552.break;}
1553.case 3:{
1554.cin.ignore();
1555.edit_menu(temp, choise);
1556.break;}
1557.case 4:{
1558.cin.ignore();
1559.delete_menu(choise, temp);
1560.return;}
1561.case 5:{
1562.return;}};}}

```

## Int main

```
1563.int main(){
1564.int choise = 0;
1565.string temp = "";
1566.int itemp = 0;
1567.SetConsoleCP(1251);
1568.SetConsoleOutputCP(1251);
1569.main_menu(choise, temp);}
```

					КР.ІП-25.00.00.000 ПЗ	Арк.
						64
Зм.	Арк.	№ докум.	Підп.	Дата		

## Додаток Б

### Результати контрольного прикладу

#### Рисунок Б.1.

Книгу знайдено!  
Код книги: 35  
Назва: 20000 льє під водою  
Жанр: роман  
Рік видання: 2011  
Видавництво: Ексмо-Україна  
Тип палітурки: М'яка  
Кількість сторінок: 464  
Мова: Українська  
Автор: Жуль Верн  
Кількість примірників: 8  
Закупівельна ціна: 68  
ціна: 102

\*\*\*Робота з книгами\*\*\*

- 1 - Пошук чи є в наявності
  - 2 - Процедура реєстрації нової книги
  - 3 - Вивід всіх книг на екран
  - 4 - Додавання надходження уже зареєстрованих книг
  - 5 - Видалення запису про книгу
- ESC - Повернення на вищий рівень

Ви вибрали "Додавання надходження вже зареєстрованої книги"

Зміна кількості книжок!  
Для збільшення кількості книжок натисніть 1, для виходу - будь-яку іншу цифру.

1  
Введіть код книги, кількість примірників якої бажаєте змінити:

35  
Скільки примірників потрібно додати?

10  
Код книги: 35  
Назва: 20000 льє під водою  
Жанр: роман  
Рік видання: 2011  
Видавництво: Ексмо-Україна  
Тип палітурки: М'яка  
Кількість сторінок: 464  
Мова: Українська  
Автор: Жуль Верн  
Кількість примірників: 18  
Закупівельна ціна: 68  
ціна: 102

Кількість змінено!

\*\*\*Робота з книгами\*\*\*

- 1 - Пошук чи є в наявності
  - 2 - Процедура реєстрації нової книги
  - 3 - Вивід всіх книг на екран
  - 4 - Додавання надходження уже зареєстрованих книг
  - 5 - Видалення запису про книгу
- ESC - Повернення на вищий рівень

Зм.	Арк.	№ докум.	Підп.	Дата

КР.ІП-25.00.00.000 ПЗ

Арк.

88

Ви вибрали "Пошук чи є в наявності"

Введіть назву книги, жанр або ім'я і прізвище автора розділяючи їх пробілом  
Місто кісток  
Книгу не знайдено!

\*\*\*Робота з книгами\*\*\*

- 1 - Пошук чи є в наявності
- 2 - Процедура реєстрації нової книги
- 3 - Вивід всіх книг на екран
- 4 - Додавання надходження уже зареєстрованих книг
- 5 - Видалення запису про книгу
- ESC - Повернення на вищий рівень

Ви вибрали "Процедура реєстрації нової книги"

Код книги: 106

Введіть інфо про нову книгу:

Назва: Місто кісток

Жанр: роман

Видавництво: Темпора

Тип палітурки: Тверда

Мова: Українська

Автор: Кассандра Клер

Рік видання: 2015

Кількість сторінок: 250

Кількість примірників: 20

Закупівельна ціна: 98

Код книги: 106

Назва: Місто кісток

Жанр: роман

Рік видання: 2015

Видавництво: Темпора

Тип палітурки: Тверда

Кількість сторінок: 250

Мова: Українська

Автор: Кассандра Клер

Кількість примірників: 20

Закупівельна ціна: 98

Ціна: 147

\*\*\*Робота з книгами\*\*\*

- 1 - Пошук чи є в наявності
- 2 - Процедура реєстрації нової книги
- 3 - Вивід всіх книг на екран
- 4 - Додавання надходження уже зареєстрованих книг
- 5 - Видалення запису про книгу
- ESC - Повернення на вищий рівень

Зм.	Арк.	№ докум.	Підп.	Дата

КР.ІП-25.00.00.000 ПЗ

Арк.

89

## Рисунок Б.2.

\*\*\*Реалізація товару\*\*\*

1 - Новий чек

ESC - Повернення на вищий рівень

Ви вибрали "Новий чек"

Пошук книги!

Введіть назву книги, жанр або ім'я і прізвище автора розділяючи їх пробілом  
Простір та історія

Книгу знайдено!

Код книги: 6

Назва: Простір та історія

Жанр: драма

Рік видання: 2013

Видавництво: Жупанського

Тип палітурки: Тверда

Кількість сторінок: 368

Мова: Українська

Автор: Бродель Фернан

Кількість примірників: 10

Закупівельна ціна: 140

Ціна: 210

Якщо пошук необхідно повторити, натисніть 0

Введіть код книги, яку потрібно додати в чек:

6

Введіть кількість примірників, які потрібно занести в чек:

1

Вартість 1 примірників книги "Простір та історія" складає 210

Для підтвердження додавання книги в чек введіть 1, для відміни - будь-яку іншу цифру

1

Книгу додано!

Для додавання ще одної книги введіть 1, для продовження формування чека - 2:

1

Введіть назву книги, жанр або ім'я і прізвище автора розділяючи їх пробілом

10 успішних українських брендів

Книгу знайдено!

Код книги: 53

Назва: 10 успішних українських брендів

Жанр: монографія

Рік видання: 2017

Видавництво: Discursus

Тип палітурки: Тверда

Кількість сторінок: 208

Мова: Українська

Автор: Борис Ославський

Кількість примірників: 10

Закупівельна ціна: 96

Ціна: 144

Якщо пошук необхідно повторити, натисніть 0

Введіть код книги, яку потрібно додати в чек:

53

Введіть кількість примірників, які потрібно занести в чек:

2

Вартість 2 примірників книги "10 успішних українських брендів" складає 288

Для підтвердження додавання книги в чек введіть 1, для відміни - будь-яку іншу цифру

1

					КР.ІП-25.00.00.000 ПЗ	Арк.
						90
Зм.	Арк.	№ докум.	Підп.	Дата		

Книгу додано!

Для додавання ще одної книги введіть 1, для продовження формування чека - 2:

2

Загальна ціна: 498

Зведіть дату продажу!

Зведіть день (дд): 02

Зведіть місяць (мм): 06

Зведіть рік (рррр): 2018

Пошук працівника!

Зведіть прізвище працівника:

Малинич

Працівника знайдено!

Код особи: 28

Прізвище: Малинич

Ім'я: Михайло

По-батькові: Васильович

Телефон: +380502456874

E-mail: malynych@i.ua

Населений пункт: Отинія

Вулиця: Сніжна

Будинок: 3

Квартира: -

Поштовий індекс: 35684

Відношення до магазину: стажер

Дата прийняття на роботу: 01/06/2018

Номер каси: 2

Дата звільнення: відсутня

Якщо пошук необхідно повторити, натисніть 0

Зведіть код продавця, якого потрібно додати в чек:

28

Якщо клієнт має дисконтну картку, введіть 1, якщо ні - 0

1

Зведіть прізвище або назву дисконту (срібний/золотий/платиновий):

Вусата

Клієнта знайдено!

Код особи: 3

Прізвище: Вусата

Ім'я: Марія

По-батькові: Юрїївна

Телефон: +380469751127

E-mail: mustache@yahoo.com

Населений пункт: Львів

Вулиця: Клепарівська

Будинок: 56

Квартира: 24

Поштовий індекс: 10254

Тип дисконту: золотий

Якщо пошук необхідно повторити, натисніть 0

Зведіть код клієнта, якого потрібно додати в чек:

3

Завершення формування чека!

6;Простір та історія;210;1;210;

53;10 успішних українських брендів;144;2;288;

2;6;2018;

28;Малинич;Михайло;2;

3;Вусата;Марія;золотий;

Сума без знижки: 498

Відсоток знижки: 15%

Сума із знижкою: 423.3

Для підтвердження операції введіть 1, для відміни - 0:

1

Ви вибрали: 1

Дані змінено!

Операція успішно завершена!

\*\*\*Реалізація товару\*\*\*

1 - Новий чек

ESC - Повернення на вищий рівень

					КР.ІП-25.00.00.000 ПЗ	Арк.
						91
Зм.	Арк.	№ докум.	Підп.	Дата		



Рисунок Б.3.

D:\DZU\A\Programming\programs\C++\kursach\MENU_prototype_v3.exe											
Як вибрати "Видати всі книги на екран"											
Код	Назва книги	Жанр	Рік	Видавництво	Пап/турка	К-ть стор.	Мова	Автор	К-ть прим.	Закуп. ціна	Ціна
1	Історія України-Русь (Не)історичні міти	драма	2009	Темпора	Тверда	344	Українська	Інна Старовойтенко	10	100	150
2	Історичні міти	драма	2017	Ніколайс клуб	Тверда	176	Українська	Володимир Г'ятович	10	100	135
3	оповідки	драма	2016	Темпора	Тверда	242	Українська	Марк Голітський	20	120	100
4	Таємна історія ополігари	драма	2016	Темпора	Тверда	312	Українська	Старт Ленсі	52	60	90
5	Картини нарядського життя, історії та природи	драма	2017	Темпора	Тверда	584	Українська	Сарген Марков	52	100	100
6	Простір та історія	драма	2013	Жуланського	Тверда	368	Українська	Бродель Фернан	11	140	210
7	Люди і речі	драма	2014	Жуланського	Тверда	232	Українська	Бродель Фернан	20	100	150
8	Історичні міти	драма	2017	Жуланського	Тверда	472	Українська	Бродель Фернан	11	150	150
9	Із покутської книги буття	драма	2012	Темпора	М'яка	580	Українська	Роман Піхмачев	23	75	112
10	Імператори Тіберія, Катігула і Клавдія	роман	2011	Каледарія	Тверда	200	Українська	Володимир Дмитренко	22	115	135
11	Істинні скарби в містах Причорномор'я	роман	2017	Наш-Час	Тверда	303	Українська	Марина Сергієвська	4	90	135
12	Awesome Beaches	роман	2017	Основи	М'яка	192	Англійська	Ганна Копилова	6	500	750
13	Awesome Oases	роман	2017	Основи	М'яка	192	Англійська	Ганна Копилова	6	500	750
14	Saracen Lawmen Przewodnik	роман	2008	Балтія-Друк	Тверда	224	Польська	Ірина Пустиннікова	30	240	360
15	Кіев Top 10	роман	2009	Балтія-Друк	Тверда	70	Французька	Вілена Ференчева	4	450	675
16	Алфавіт-путівник: Україна для виховців пригод	роман	2010	Друтбів	Тверда	268	Українська	Вілена Космова	13	100	150
17	Замки і фортеці західної України	нарис	2008	Центр Європи	М'яка	200	Українська	Оксана Майж	30	190	195
18	західна Україна: Путівник	нарис	2008	Центр Європи	М'яка	288	Російська	Сергійс Задомік	20	200	230
19	Step By Step: Київ	нарис	2016	Центр Європи	Тверда	264	Українська	Вілена Космова	32	120	180
20	Пам'ятники національного культурного насліддя України	роман	2012	Балтія-Друк	Тверда	288	Російська	Ірина Лівіно	10	90	135
21	Антологія українського міфу: Потоки	роман	2017	Балтія-Друк	Тверда	848	Українська	Валерія Мельничук	24	300	450
22	Вино і філософія. Символізм думки і келиха	байка	2010	Темпора	Тверда	368	Українська	Фріц Олф	15	56	84
23	Грехи, арабі і ми	нарис	2012	Ніка-Центр	Тверда	304	Українська	Франц Ботгін	26	320	480
24	Сучасна літературна комуністична: Стратегії і методи Антології	байка	2017	Ніка-Центр	Тверда	488	Українська	Дмитро Волковик	28	120	180
25	Сеген Міхаленко: людина на тлі епохи	байка	2009	Темпора	Тверда	544	Українська	Інна Старовойтенко	10	150	225
26	Історія та сучасна гуманітаристика: дослідження з теорії знання про минуле	байка	2012	Ніка-Центр	Тверда	264	Українська	Елена Доманська	25	100	150
27	Ніка-Центр	байка	2016	Радуга	Тверда	255	Українська	Костянтин Родик	27	114	171
28	Нариси з соціології	байка	2010	ІМА	М'яка	243	Українська	Лариса Масенко	47	320	480
29	Знання мови – одним словом	байка	2005	Артпрі	Тверда	420	Українська	Тарас Береза	26	120	180
30	12 капелів успіху	байка	2017	Старого Лева	Тверда	208	Українська	Стівен Кові	2	210	315
31	Ідеальний командний гравець. Як розпізнати і розвинути три основні якості	байка	2017	КСД	Тверда	196	Українська	Патрік Ленсіоні	12	126	144
32	Нариси з української літературної мови та її лінгвістичної структури	байка	2012	Темпора	Тверда	664	Українська	Врій Шовалів	31	250	360
33	365 пригід на щодень	байка	2013	Снічудо	Тверда	336	Українська	Діана Віссон	26	62	93
34	103 думати	байка	2010	Старого Лева	Тверда	240	Українська	Домі Селі	20	100	150
35	20000 лів під водою	роман	2011	Ексмо-Україна	М'яка	464	Українська	Жуль Верн	8	68	102
36	28 днів із життя бурдука	казка	2017	Фонтан казок	Тверда	576	Українська	Інна Андрусів	24	96	144
37	16 і 6 котів-детективів	казка	2017	Старого Лева	Тверда	144	Українська	Ганна Доманченко	21	100	150
38	Jetzt kam ich aufs Klo	роман	2008	G & G Publishing House	Тверда	24	Німецька	Сюзі Гаммерле	5	290	435
39	Абетка	казка	2011	Старого Лева	Тверда	264	Українська	Оксана Кротків	12	145	216
40	Барачник Раселл і забулений скарб	казка	2013	ВСЛ	Тверда	404	Українська	Робі Скінтон	40	126	176
41	В язень замку Палакко	казка	2013	Карпатська вежа	Тверда	368	Українська	Мірослав Доночич	1	159	238
42	Великий Дружний Белегет	казка	2016	A-1-a-B-a-1-a-MA-MA-1-a	Тверда	272	Українська	Дар'яна Діа	12	48	72
43	Ілюстрована енциклопедія України	нарис	2008	Балтія-Друк	Тверда	160	Українська	Врій Шовалів	5	90	135
44	Київ 1917-1919	роман	2008	Темпора	Тверда	136	Українська	Олександр Кучерук	24	100	150
45	Ілюстрована енциклопедія України	роман	2008	Ілюстрація	Тверда	240	Українська	Кристіна Селі	20	100	150
46	Сучасна Україна: портрет з натури	нарис	2011	Поліграфіцентр Ліра	М'яка	296	Українська	Сергієвський	12	98	147
47	Рух опору в Україні: 1960-1990	нарис	2011	Смолюсик	Тверда	804	Українська	Оксана Зізнавич	10	100	150
48	Атлас Бєсєву для дітей	байка	2014	БАО	Тверда	604	Українська	Володимир Меников	10	50	75
49	Атлас тварин і рослин України	байка	2014	Талант	Тверда	64	Українська	Тетяна Станкевич	21	50	75
50	Нова енциклопедія для дітей	байка	2013	Роберт Коп	Тверда	62	Українська	Роберт Коп	20	20	30
51	Світ тварин весті континентів	нарис	2014	БАО	Тверда	64	Українська	Олексій Оксєнов	10	50	75
52	Чи може Бєсєво вклати Бєсєву?	байка	2016	КСД	Тверда	256	Українська	Дієнна Гарріс	23	94	138
53	100 успішних українських Бєсєв	моніграфія	2017	Discussus	Тверда	208	Українська	Борис Шовалів	1	66	99
54	Rework	моніграфія	2016	Клуб сімейного дозвілля	Тверда	176	Українська	Дієнна Дієнна	4	200	300
55	Чи книга змігати все погляд на Бєсєво	моніграфія	2016	Клуб сімейного дозвілля	Тверда	176	Українська	Дієнна Дієнна	25	375	562
56	Книга змігати все погляд на Бєсєво	моніграфія	2017	Lakobro Publishing	Тверда	488	Українська	Ферріс Скотт	24	300	450
57	Історія Старбєсєво. Все погляд на Бєсєво	моніграфія	2017	Наш формат	Тверда	408	Українська	Горька Ворд	26	350	525
58	214 нуд на доріжці	моніграфія	2017	Наш формат	Тверда	232	Українська	Пєтер Ворд	24	300	450
59	Велика миста	моніграфія	2016	ВСЛ	Тверда	254	Українська	Елзабет Гілберт	2	146	219
60	Вузити-Бутти. Історія Кієво, розказана її засновником	моніграфія	2017	Наш формат	Тверда	432	Українська	Елзабет Гілберт	21	245	368
61	Влада і контроль у добу глобалізації: Нова світова політична економія	моніграфія	2017	Ніка-Центр	Тверда	385	Українська	Бек Уліхріс	28	240	360
62	Еволюція грошей, фінансова історія світу	моніграфія	2017	Наш формат	Тверда	384	Українська	Нік Фергісон	24	256	384
63	Влада і контроль у добу глобалізації: Нова світова політична економія	моніграфія	2017	Ніка-Центр	Тверда	405	Українська	Бек Уліхріс	28	240	360

1	D:\UA\AP\Programming\programs\C++\kursach\MENU_prototype_v3.exe	roman	2008	G & G Publishing House	Тверда	27	Пічменська	Олена Гаммерль	5	230	431
38	Abetka	казка	2011	Старого Лева	Тверда	24	Українська	Оксана Кротюк	12	95	147
40	барачник Рассел і загублений казуб	казка	2017	BSL	Тверда	40	Українська	Роб Скоттон	6	126	189
42	Важкі замки Паломо	казка	2013	Паризьська вежа	Тверда	359	Українська	Мірослава Демиченя	159	99	142
43	Великий Дружний Велетень	казка	2016	A-BA-BA-GA-LA-MA-GA	Тверда	272	Українська	Дід Роланд	4	124	186
44	Дісторична енциклопедія України	нарис	2008	Балтія-Друк	Тверда	160	Українська	Мрія Шопал	5	90	135
45	Миті 1917-1919	нарис	2008	Темпора	Тверда	366	Українська	Олександр Кучерук	10	100	150
46	Нанонаука. Невидима революція	роман	2008	Иностранка	Тверда	240	Російська	Крістіан Хоакін	20	100	150
47	Сучасна Україна: портрет з натури	нарис	2013	Поліграфцентр Ліра	М'яка	96	Українська	Сергій Федюка	20	100	150
48	Дух опору України: 1980-1990	нарис	2010	Моноліт	Тверда	804	Українська	Оксана Шинкарук	10	120	180
49	Атлас ієсвітлиць для дітей	нарис	2014	БАО	Тверда	64	Українська	Володимир Шеніков	10	50	75
50	Атлас тварин і рослин України	нарис	2016	Тарант	Тверда	60	Українська	Тетяна Гаммерль	21	100	150
51	Нова енциклопедія для дитини	нарис	2013	Роберт Копл	Тверда	192	Російська	Роберт Копл	2	100	150
52	Світ тварин шести континентів	нарис	2014	БАО	Тверда	64	Українська	Олексій Осесов	10	50	75
53	Чи може баскетол стати баскетом?	нарис	2016	КСД	Тверда	146	Українська	Діагона Сторі	12	100	150
54	10 успішних українських брендів	монографія	2017	Discussum	Тверда	208	Українська	Борис Олександрів	1	96	144
55	Renew	монографія	2016	Клуб сімейного дозвілля	Тверда	176	Українська	В'ячеслав Федюк	216	1	300
56	Чи мінга змінився ваш погляд на бізнес	монографія	2016	Клуб сімейного дозвілля	Тверда	176	Українська	В'ячеслав Федюк	25	250	375
57	Інга	монографія	2017	Yakaboo Publishing	Тверда	408	Українська	Дарія Захаревич	24	300	450
58	Історія Starbuck's. Усе почалося з чашки кави...	монографія	2017	Наш формат	Тверда	408	Українська	Гомаріа Флор	26	350	525
59	Вілка муля до одиниці	монографія	2016	Наш формат	Тверда	232	Українська	Петро Тінь	23	200	300
60	Беліка-Бугія	монографія	2016	BSL	Тверда	254	Українська	Елізабет Гіберт	2	146	219
61	Вулиці-Богаті. Історія Німеч, розказана її засновником	монографія	2017	Наш формат	Тверда	432	Українська	Віт Найд	21	245	367
62	Влада і контролер у добу глобалізації: Нова світова політична економія	монографія	2013	Ніка-Центр	Тверда	405	Українська	Бек Ульріх	28	240	360
63	Еволюція транснаціональної економіки світу	монографія	2016	Наш формат	Тверда	384	Українська	Ніа Феррерос	28	356	534
64	Економіка: інструменти з використання	монографія	2017	Наш формат	Тверда	384	Українська	Ха Найд	20	200	300
65	A taste of life	роман	2012	ACS	Тверда	296	Англійська	Мілош Воярський	30	200	300
66	Amn rufm alle Horrelporr	роман	2017	G & G Publishing House	Тверда	32	Українська	Міра Найд	9	100	150
67	Hearts of Three	роман	2017	Знання	Тверда	359	Англійська	Джекс Лондон	10	600	900
68	PS Я люблю тебе	роман	2010	Иностранка	Тверда	496	Російська	Сесілія Аєрлі	19	148	222
69	Абсолютно неімовірно	повість	2013	Старого Лева	Тверда	320	Українська	Нічиріа Гринштейн	14	100	150
70	Старий Лев	повість	2016	Махаон	Тверда	512	Українська	Ойган Магган	12	152	228
71	Душа: Мертвий сезон	повість	2013	УБІНО	Тверда		Українська	Барбара Космоєвська	124	13	199
72	Вулиці-Богаті. Історія Німеч, розказана її засновником	повість	2012	Порткан	Тверда	96	М'яка	Нічиріа Гринштейн	14	164	244
73	Душа: Мертвий сезон	повість	2007	A-BA-BA-GA-LA-MA-GA	Тверда	432	Українська	Володимир Туртківський	16	144	216
74	Гаррі Поттер і філософський камінь	повість	2001	A-BA-BA-GA-LA-MA-GA	Тверда	320	Українська	Діагона Ролінг	20	120	180
75	Гаррі Поттер і таємна кімната	повість	2004	A-BA-BA-GA-LA-MA-GA	Тверда	352	Українська	Діагона Ролінг	14	125	187
76	Гаррі Поттер і в'язень Азкабана	повість	2006	A-BA-BA-GA-LA-MA-GA	Тверда	316	Українська	Діагона Ролінг	18	125	187
77	Гаррі Поттер і мовка вогню	повість	2003	A-BA-BA-GA-LA-MA-GA	Тверда	672	Українська	Діагона Ролінг	18	125	187
78	Гаррі Поттер і орден фенікса	повість	2003	A-BA-BA-GA-LA-MA-GA	Тверда	816	Українська	Діагона Ролінг	20	125	187
79	Гаррі Поттер і полонізований принц	повість	2007	A-BA-BA-GA-LA-MA-GA	Тверда	840	Українська	Діагона Ролінг	14	125	187
80	Гаррі Поттер і смертні реліквії	повість	2007	A-BA-BA-GA-LA-MA-GA	Тверда	816	Українська	Діагона Ролінг	12	125	187
81	Гаррі Поттер і прокляте дитя	повість	2016	A-BA-BA-GA-LA-MA-GA	Тверда	352	Українська	Діагона Ролінг	30	200	300
82	96 м'б в поросіа о темеріа оіа	монографія	2016	Моноліт	Тверда	160	Російська	Сатен Комаровський	10	100	150
83	Ватгітніі і пологи в запитаннях і відповідях	монографія	2012	Богдан	М'яка	264	Українська	Валерія Федюка	2	120	180
84	Вігнірніа Історія	есе	2015	КСД	Тверда	288	Українська	Діагона Ролінг	20	100	150
85	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
86	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
87	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
88	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
89	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
90	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
91	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
92	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
93	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
94	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
95	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
96	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
97	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
98	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
99	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
100	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
101	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
102	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
103	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
104	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
105	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
106	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
107	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
108	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
109	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
110	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
111	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
112	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
113	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
114	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
115	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
116	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
117	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
118	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
119	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
120	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
121	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
122	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
123	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
124	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
125	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
126	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
127	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
128	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
129	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
130	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
131	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
132	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
133	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
134	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
135	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
136	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
137	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
138	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
139	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
140	Вігнірніа Історія: рз: постіиіа до розсудливіа батьків	есе	2015	Моноліт	Тверда	640	Українська	Сатен Комаровський	10	100	150
141	Вігнірніа Історія: рз										

#### Рисунок Б.4.

Ви вибрали "Робота з записами про працівників"

\*\*\*Робота з записами про працівників\*\*\*

- 1 - Пошук
- 2 - Реєстрація нового працівника
- 3 - Редагування вже зареєстрованого працівника
- 4 - Вивід усіх працівників на екран
- ESC - Повернення на вищий рівень

Ви вибрали "Реєстрація нового працівника"

Введіть інфо про нового працівника:

Код особи: 28

Прізвище: Малинич

Ім'я: Михайло

По-батькові: Васильович

Телефон: +380502456874

E-mail: malynych@i.ua

Населений пункт: Отинія

Вулиця: Сніжна

Будинок: 3

Квартира: -

Поштовий індекс: 35684

Відношення до магазину: стажер

Дата прийняття на роботу (дд/мм/рррр): 01/06/2018

Номер каси: 2

Дата звільнення (дд/мм/рррр): відсутня

Код особи: 28

Прізвище: Малинич

Ім'я: Михайло

По-батькові: Васильович

Телефон: +380502456874

E-mail: malynych@i.ua

Населений пункт: Отинія

Вулиця: Сніжна

Будинок: 3

Квартира: -

Поштовий індекс: 35684

Відношення до магазину: стажер

Дата прийняття на роботу: 01/06/2018

Номер каси: 2

Дата звільнення: відсутня

\*\*\*Робота з записами про працівників\*\*\*

- 1 - Пошук
- 2 - Реєстрація нового працівника
- 3 - Редагування вже зареєстрованого працівника
- 4 - Вивід усіх працівників на екран
- ESC - Повернення на вищий рівень

					КР.ІП-25.00.00.000 ПЗ	Арк.
						93
Зм.	Арк.	№ докум.	Підп.	Дата		

## Рисунок Б.5.

Пошук працівника!  
Введіть прізвище працівника:  
Лучка

Працівника знайдено!  
Код особи: 27  
Прізвище: Лучка  
Ім'я: Андрій  
По-батькові: Іванович  
Телефон: +380995678412  
E-mail: luch.luch@gmail.com  
Населений пункт: Калуш  
Вулиця: Хіміків  
Будинок: 20  
Квартира: 1  
Поштовий індекс: 15675  
Відношення до магазину: стажер  
Дата прийняття на роботу: 01/06/2018  
Номер каси: 1  
Дата звільнення: відсутня

Редагування даних працівника!  
Для редагування натисніть 1, для виходу - будь-яку іншу цифру.  
1  
Введіть код працівника, дані якого бажаєте змінити:  
27

Ви обрали для редагування запис про клієнта:  
Код особи: 27  
Прізвище: Лучка  
Ім'я: Андрій  
По-батькові: Іванович  
Телефон: +380995678412  
E-mail: luch.luch@gmail.com  
Населений пункт: Калуш  
Вулиця: Хіміків  
Будинок: 20  
Квартира: 1  
Поштовий індекс: 15675  
Відношення до магазину: стажер  
Дата прийняття на роботу: 01/06/2018  
Номер каси: 1  
Дата звільнення: відсутня

Яке поле бажаєте відредагувати?

1-Прізвище  
2-Ім'я  
3-По-батькові  
4-Телефон  
5-E-mail  
6-Населений пункт  
7-Вулиця  
8-Будинок  
9-Квартира  
a-Поштовий індекс  
b-Відношення до магазину  
c-Дата звільнення  
d-Номер каси  
z-Завершення редагування та занесення змін до бази даних

Введіть символ Вашого вибору:

					КР.ІП-25.00.00.000 ПЗ	Арк.
						94
Зм.	Арк.	№ докум.	Підп.	Дата		

Ви вибрали: с

Введіть нову дату звільнення:

08/06/2018

Яке поле бажаєте відредагувати?

1-Прізвище

2-Ім'я

3-По-батькові

4-Телефон

5-E-mail

6-Населений пункт

7-Вулиця

8-Будинок

9-Квартира

a-Поштовий індекс

b-Відношення до магазину

c-Дата звільнення

d-Номер каси

z-Завершення редагування та занесення змін до бази даних

Введіть символ Вашого вибору:

Ви вибрали: b

Введіть нове відношення до магазину:

звільнений продавець

Яке поле бажаєте відредагувати?

1-Прізвище

2-Ім'я

3-По-батькові

4-Телефон

5-E-mail

6-Населений пункт

7-Вулиця

8-Будинок

9-Квартира

a-Поштовий індекс

b-Відношення до магазину

c-Дата звільнення

d-Номер каси

z-Завершення редагування та занесення змін до бази даних

Введіть символ Вашого вибору:

Ви вибрали: z

Код особи: 27

Прізвище: Лучка

Ім'я: Андрій

По-батькові: Іванович

Телефон: +380995678412

E-mail: luch.luch@gmail.com

Населений пункт: Калуш

Вулиця: Хіміків

Будинок: 20

Квартира: 1

Поштовий індекс: 15675

Відношення до магазину: звільнений продавець

Дата прийняття на роботу: 01/06/2018

Номер каси: 1

Дата звільнення: 08/06/2018

Дані змінено!

					КР.ІП-25.00.00.000 ПЗ	Арк.
						95
Зм.	Арк.	№ докум.	Підп.	Дата		

## Рисунок Б.6.

Пошук працівника!  
Введіть прізвище працівника:  
Соботник

Працівника знайдено!  
Код особи: 11  
Прізвище: Соботник  
Ім'я: Христина  
По-батькові: Віталіївна  
Телефон: +380663719056  
E-mail: crissy@gmail.com  
Населений пункт: Івано-Франківськ  
Вулиця: Бельведерська  
Будинок: 23  
Квартира: 85  
Поштовий індекс: 76014  
Відношення до магазину: стажер  
Дата прийняття на роботу: 01/05/2018  
Номер каси: 1  
Дата звільнення: відсутня

Редагування даних працівника!  
Для редагування натисніть 1, для виходу - будь-яку іншу цифру.  
1  
Введіть код працівника, дані якого бажаєте змінити:  
11

Ви обрали для редагування запис про клієнта:  
Код особи: 11  
Прізвище: Соботник  
Ім'я: Христина  
По-батькові: Віталіївна  
Телефон: +380663719056  
E-mail: crissy@gmail.com  
Населений пункт: Івано-Франківськ  
Вулиця: Бельведерська  
Будинок: 23  
Квартира: 85  
Поштовий індекс: 76014  
Відношення до магазину: стажер  
Дата прийняття на роботу: 01/05/2018  
Номер каси: 1  
Дата звільнення: відсутня

Яке поле бажаєте відредагувати?

1-Прізвище  
2-Ім'я  
3-По-батькові  
4-Телефон  
5-E-mail  
6-Населений пункт  
7-Вулиця  
8-Будинок  
9-Квартира  
a-Поштовий індекс  
b-Відношення до магазину  
c-Дата звільнення  
d-Номер каси  
z-Завершення редагування та занесення змін до бази даних

Введіть символ Вашого вибору:

					КР.ІП-25.00.00.000 ПЗ	Арк.
						96
Зм.	Арк.	№ докум.	Підп.	Дата		



Введіть символ Вашого вибору:  
Ви вибрали: b

Введіть нове відношення до магазину:  
чинний продавець  
Яке поле бажаєте відредагувати?

1-Прізвище  
2-Ім'я  
3-По-батькові  
4-Телефон  
5-E-mail  
6-Населений пункт  
7-Вулиця  
8-Будинок  
9-Квартира  
a-Поштовий індекс  
b-Відношення до магазину  
c-Дата звільнення  
d-Номер каси  
z-Завершення редагування та занесення змін до бази даних

Введіть символ Вашого вибору:  
Ви вибрали: z

Код особи: 11  
Прізвище: Сobotник  
Ім'я: Христина  
По-батькові: Віталіївна  
Телефон: +380663719056  
E-mail: crissy@gmail.com  
Населений пункт: Івано-Франківськ  
Вулиця: Бельведерська  
Будинок: 23  
Квартира: 85  
Поштовий індекс: 76014  
Відношення до магазину: чинний продавець  
Дата прийняття на роботу: 01/05/2018  
Номер каси: 1  
Дата звільнення: відсутня  
  
Дані змінено!

					КР.ІП-25.00.00.000 ПЗ	Арк.
						97
Зм.	Арк.	№ докум.	Підп.	Дата		

## Рисунок Б.7.

Ви вибрали "Робота з записами про клієнтів"

\*\*\*Робота з записами про клієнтів\*\*\*

- 1 - Пошук
- 2 - Реєстрація нового клієнта
- 3 - Редагування даних уже зареєстрованого клієнта
- 4 - Вивід усіх клієнтів на екран
- ESC - Повернення на вищий рівень

Ви вибрали "Реєстрація нового клієнта"

Введіть інфо про нового клієнта:

Код особи: 29  
Прізвище: Драгомирецький  
Ім'я: Петро  
По-батькові: Петрович  
Телефон: +380996523471  
E-mail: dragomyr.petya@yahoo.com  
Населений пункт: Івано-Франківськ  
Вулиця: Чорновола  
Будинок: 50  
Квартира: 14  
Поштовий індекс: 15487  
Тип дисконту: срібний

Код особи: 29  
Прізвище: Драгомирецький  
Ім'я: Петро  
По-батькові: Петрович  
Телефон: +380996523471  
E-mail: dragomyr.petya@yahoo.com  
Населений пункт: Івано-Франківськ  
Вулиця: Чорновола  
Будинок: 50  
Квартира: 14  
Поштовий індекс: 15487  
Тип дисконту: срібний

\*\*\*Робота з записами про клієнтів\*\*\*

- 1 - Пошук
- 2 - Реєстрація нового клієнта
- 3 - Редагування даних уже зареєстрованого клієнта
- 4 - Вивід усіх клієнтів на екран
- ESC - Повернення на вищий рівень

					КР.ІП-25.00.00.000 ПЗ	Арк.
						98
Зм.	Арк.	№ докум.	Підп.	Дата		

## Рисунок Б.8

\*\*\*MENU\*\*\*

- 1 - Робота з книгами
- 2 - Реалізація товару
- 3 - Робота з записами про клієнтів
- 4 - Звітність по роботі магазину
- 5 - Робота з записами про працівників
- ESC - Exit program

Ви вибрали "Звітність по роботі магазину"

\*\*\*Звітність по роботі магазину\*\*\*

- 1 - Топ продажі за вибраний місяць
- 2 - Топ клієнти за вибраний місяць
- 3 - Топ працівники за вибраний місяць
- 4 - Телефони працівників
- ESC - Повернення на вищий рівень

Ви вибрали "Топ продажі за вибраний місяць"

Введіть місяць і рік за якими необхідно сформувати звіт:

Місяць: 05

Рік: 2018

Топ продажі за 5.2018

№	Назва книги	К-ть
1	Гаррі Поттер і смертельні реліквії	6
2	Гаррі Поттер і напівкровний принц	5
3	Відьмак. Останнє бажання	3
4	Місто кісток	2
5	Відьмак. Хрещення вогнем	2
6	A taste of life	2
7	Греки, араби і ми	1
8	Відьмак. Сезон гроз	1
9	Hearts of Three	1
10	PS Я люблю тебе	1
11	Західна Україна. Путівник	1

\*\*\*Звітність по роботі магазину\*\*\*

- 1 - Топ продажі за вибраний місяць
- 2 - Топ клієнти за вибраний місяць
- 3 - Топ працівники за вибраний місяць
- 4 - Телефони працівників
- ESC - Повернення на вищий рівень



## Рисунок Б.9.

Ви вибрали "Вивід всіх клієнтів на екран"

Код особи: 10  
Прізвище: Саботнік  
Ім'я: Марія  
По-батькові: Віталіївна  
Телефон: +380663719077  
E-mail: sabatnik@gmail.com  
Населений пункт: Київ  
Вулиця: Млинарська  
Будинок: 15  
Квартира: 6  
Поштовий індекс: 25630  
Тип дисконту: срібний

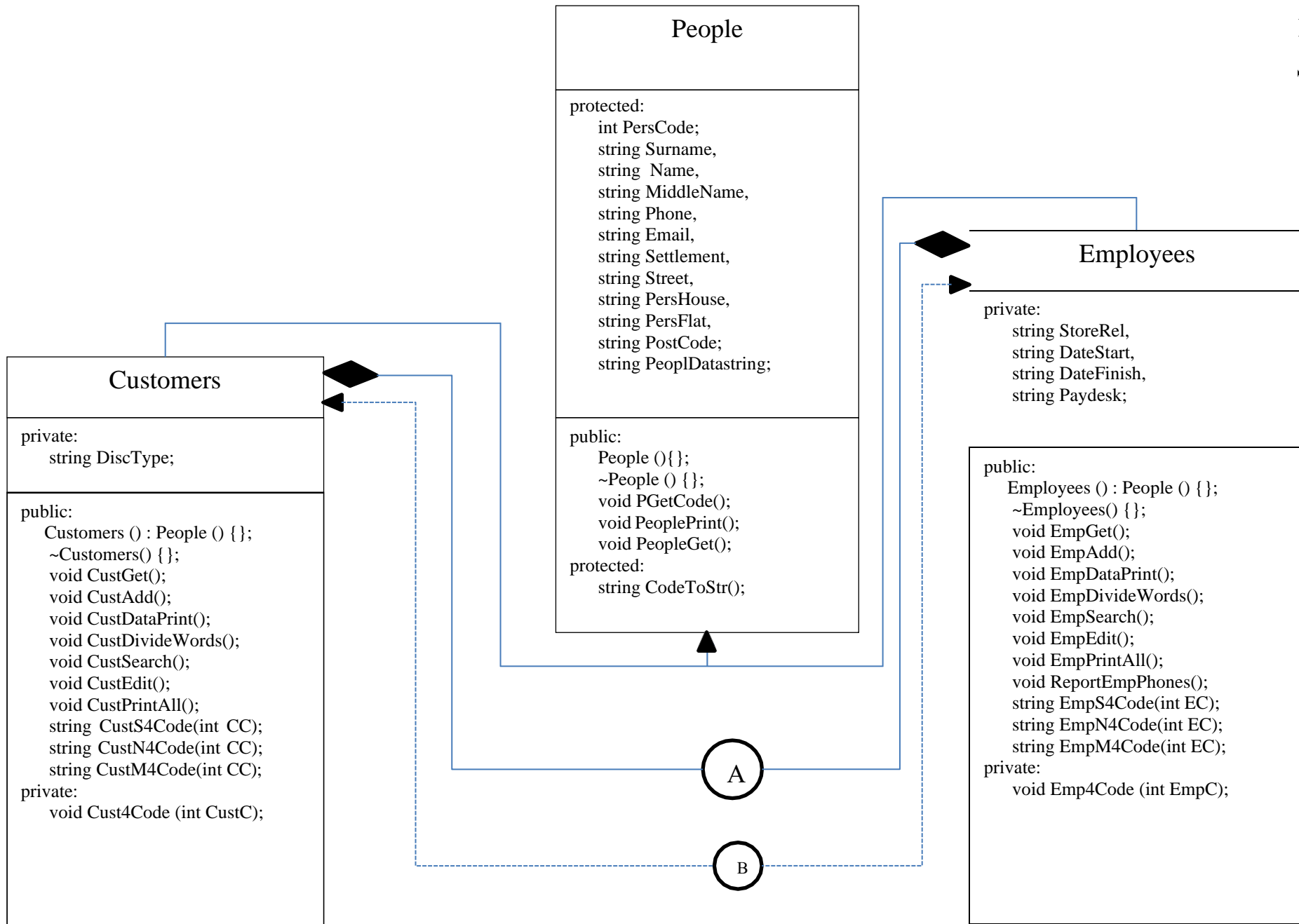
Код особи: 1  
Прізвище: Тимків  
Ім'я: Галина  
По-батькові: Йосипівна  
Телефон: +380456842122  
E-mail: tymkivg@ukr.net  
Населений пункт: Луцьк  
Вулиця: Малинівська  
Будинок: 55  
Квартира: 12  
Поштовий індекс: 25104  
Тип дисконту: золотий

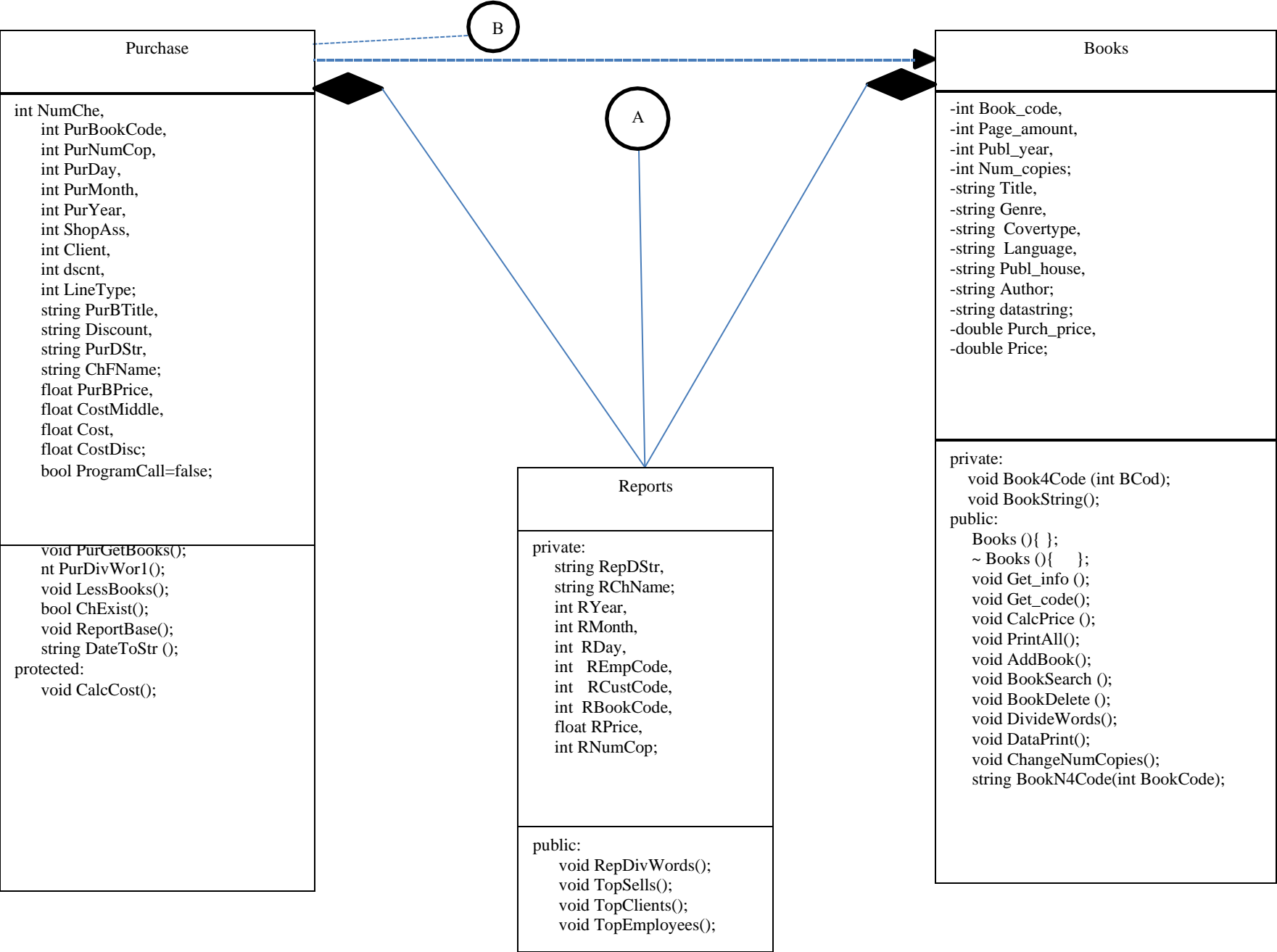
Код особи: 2  
Прізвище: Козлова  
Ім'я: Вікторія  
По-батькові: Іванівна  
Телефон: +380544816548  
E-mail: koziol@i.ua  
Населений пункт: Харків  
Вулиця: Східна  
Будинок: 22  
Квартира: 11  
Поштовий індекс: 15644  
Тип дисконту: золотий

Код особи: 3  
Прізвище: Вусата  
Ім'я: Марія  
По-батькові: Юріївна  
Телефон: +380469751127  
E-mail: mustache@yahoo.com  
Населений пункт: Львів  
Вулиця: Клепарівська  
Будинок: 56  
Квартира: 24  
Поштовий індекс: 10254  
Тип дисконту: золотий

Код особи: 4  
Прізвище: Галюк  
Ім'я: Ярема  
По-батькові: Степанович  
Телефон: +380996547821  
E-mail: galiuk@gmail.com

					КР.ІП-25.00.00.000 ПЗ	Арк.
						100
Зм.	Арк.	№ докум.	Підп.	Дата		





					КР.ІП-25.00.00.000 ПЗ	Арк.
						103
Зм.	Арк.	№ докум.	Підп.	Дата		