

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ  
КАФЕДРА ПРОГРАМУВАННЯ ТА МАТЕМАТИКИ

## **Пояснювальна записка**

до дипломної роботи

**бакалавр**

(освітньо-кваліфікаційний рівень)

**на тему «Розробка автоматизованої інформаційної системи обліку товару  
мережі аптек»**

Виконав: студент 4 курсу, групи ІТ-141  
напряму підготовки 6.040302 „Інформатика”  
спеціальності 7.04030201 „Інформатика”

\_\_\_\_\_ Пяташ Д.П.  
(підпис)

Керівник,  
доцент, к.т.н. \_\_\_\_\_ Іванов В.Г.  
(підпис)

Рецензент,  
доцент, к.т.н. \_\_\_\_\_ Фесенко Т.М.  
(підпис)

СЄВЕРОДОНЕЦЬК  
2018 року

Факультет інформаційних технологій та електроніки

Кафедра програмування та математики

Освітньо-кваліфікаційний рівень бакалавр

Напрямок підготовки 6.040302 „Інформатика”

Спеціальність 7.04030201 „Інформатика”

ЗАТВЕРДЖУЮ

Завідувач кафедри ПМ,

к.т.н., доцент

\_\_\_\_\_ Лифар В.

«\_\_\_» \_\_\_\_\_ 2018 р.

## З А В Д А Н Н Я

НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Пяташа Дмитро Петровича

### **1. Тема роботи Розробка автоматизованої інформаційної системи обліку товару мережі аптек.**

керівник роботи **доцент Іванов Віталій Геннадійович**

затверджені наказом вищого навчального закладу від “\_\_\_” \_\_2018 року № \_\_\_

2. Строк подання студентом роботи 20 травня 2018 р.

3. Вихідні дані до роботи

Об'єктом даної роботи є розробка автоматизованої інформаційної системи обліку товару мережі аптек.

3.1 Літературні джерела:

Бастриков М.В. Информационные технологии управления:– Калининград: Изд-во Ин-та «КВШУ», 2005.

Рудаков А. В. Технология разработки программных продуктов:– М.: ИЦ «Академия»; 2012.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

4.1 Вступ

4.2 Аналіз предметної галузі (огляд літератури), з висвітленням наступних питань:

Тенденції та перспективи автоматизованої інформаційної системи обліку товару мережі аптек.

Аналіз ринку програмних продуктів.

4.3 Основна частина, в якій висвітлити:

Інформаційна модель об'єкту.

Інформаційна система діяльності мережі аптек.

Опис розробленої комп'ютерної програми.

4.4 Висновки

4.4 Перелік використаних джерел

5. Перелік графічного матеріалу немає

6. Дата видачі завдання 1 лютого 2018 року.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Одержання завдання на виконання роботи	01.02.18	
2	Укладання і погодження з керівником плану і етапів виконання роботи	20.02.18	
3	Узагальнення даних літературних джерел, укладання розділу «Аналіз предметної галузі»	1.03.18	
4	Аналіз шляхів виконання завдання. Вибір і погодження з керівником оптимального шляху	10.03.18	
3	Проектування інфологічної моделі задачі що реалізується.	01.04.18	
5	Укладання та тестування програмного продукту	20.04.18	
6	Укладання, оформлення та погодження пояснювальної записки з керівником	10.05.18	
7	Укладання, оформлення та погодження з консультантом розділу «Охорона праці»	15.05.18	
7	Здача готової пояснювальної записки на кафедрі	20.05.18	
8	Укладання доповіді і презентації	01.06.18	

Студент

\_\_\_\_\_  
(підпис)

Керівник роботи

\_\_\_\_\_  
(підпис)

Факультет інформаційних технологій та електроніки

Кафедра програмування та математики

Освітньо-кваліфікаційний рівень бакалавр

Напрямок підготовки 6.040302 „Інформатика”

Спеціальність 7.04030201 „Інформатика”

ЗАТВЕРДЖУЮ

Завідувач кафедри ПМ,

к.т.н., доцент

\_\_\_\_\_ Лифар В.

«\_\_\_» \_\_\_\_\_ 2018 р.

## З А В Д А Н Н Я

НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Пяташа Дмитро Петровича

### **1. Тема роботи Розробка автоматизованої інформаційної системи обліку товару мережі аптек.**

керівник роботи доцент Іванов Віталій Геннадійович

затверджені наказом вищого навчального закладу від “\_\_\_” \_\_2018 року № \_\_\_

2. Строк подання студентом роботи 20 травня 2018 р.

3. Вихідні дані до роботи

Об'єктом даної роботи є розробка автоматизованої інформаційної системи обліку товару мережі аптек.

3.1 Літературні джерела:

Бастриков М.В. Информационные технологии управления:– Калининград: Изд-во Ин-та «КВШУ», 2005.

Рудаков А. В. Технология разработки программных продуктов:– М.: ИЦ «Академия»; 2012.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

4.1 Вступ

4.2 Аналіз предметної галузі (огляд літератури), з висвітленням наступних питань:

Тенденції та перспективи автоматизованої інформаційної системи обліку товару мережі аптек.

Аналіз ринку програмних продуктів.

4.3 Основна частина, в якій висвітлити:

Інформаційна модель об'єкту.

Інформаційна система діяльності мережі аптек.

Опис розробленої комп'ютерної програми.

4.4 Висновки

4.4 Перелік використаних джерел

5. Перелік графічного матеріалу немає

6. Дата видачі завдання 1 лютого 2018 року.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Одержання завдання на виконання роботи	01.02.18	
2	Укладання і погодження з керівником плану і етапів виконання роботи	20.02.18	
3	Узагальнення даних літературних джерел, укладання розділу «Аналіз предметної галузі»	1.03.18	
4	Аналіз шляхів виконання завдання. Вибір і погодження з керівником оптимального шляху	10.03.18	
3	Проектування інфологічної моделі задачі що реалізується.	01.04.18	
5	Укладання та тестування програмного продукту	20.04.18	
6	Укладання, оформлення та погодження пояснювальної записки з керівником	10.05.18	
7	Укладання, оформлення та погодження з консультантом розділу «Охорона праці»	15.05.18	
7	Здача готової пояснювальної записки на кафедрі	20.05.18	
8	Укладання доповіді і презентації	01.06.18	

Студент

\_\_\_\_\_  
(підпис)

Керівник роботи

\_\_\_\_\_  
(підпис)

## РЕФЕРАТ

Текст – 70., рис. – 7, табл. – 1, додатків – 1, літературних джерел – 10

Метою роботи - є розробка автоматизованої інформаційної системи обліку товару мережі аптек.

За допомогою Delphi XE7 на базі середовища Embarcadero RAD Studio XE7 розроблена комп'ютерна програма, яка дозволяє поліпшити роботу мережі аптек з обліку і реалізації товару .

Комп'ютерна програма виконує наступні основні функції:

- інформація про наявність ліків в певній аптеці;
- редагувати інформацію про наявність ліків в певній аптеці;
- загальну вартість всіх ліків у всіх аптеках;
- за назвою ліки видає список аптек, в яких воно є;
- знаходить за запитом найдешевші ліки.

ІНФОРМАЦІЙНА СИСТЕМА, МЕРЕЖА АПТЕК, DELPHI

## ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД.....	8
1.1 Необхідність автоматизації .....	10
<b>1.2 Які інформаційні системи потрібні аптеці</b> .....	13
<b>1.3 Тенденції та перспективи</b> .....	16
1.4 Аналіз ринку програмних продуктів .....	18
РОЗДІЛ 2 ІНФОРМАЦІЙНА МОДЕЛЬ.....	22
2.1 Інформаційна модель об'єкту .....	22
2.2 Інформаційна система діяльності мережі аптек.....	26
2.3 Об'єкт автоматизації в системі організації мережі аптек.....	29
РОЗДІЛ 3 ОПИС ПРОГРАМИ .....	38
3.1 Опис середовища програмування .....	34
<b>3.2 Опис програми</b> .....	40
<b>3.3 Обробка помилок в програмі</b> .....	43
ВИСНОВКИ.....	48
СПИСОК ЛІТЕРАТУРИ.....	49
ДОДАТОК А.....	51

Актуальність досліджень. На сьогоднішній день у світі інформаційні технології досягли високого рівня розвитку. У зв'язку з цим деякі мережі аптек використовують автоматичні засоби, що дозволяють ефективно зберігати, обробляти і розподіляти накопичені дані. Виходячи з сучасних вимог, що пред'являються до якості роботи підприємства, не можна не відзначити, що їх ефективна робота цілком залежить від рівня оснащення інформаційними засобами на базі комп'ютерних систем автоматизованого обліку.

Управлінська діяльність виступає в сучасних умовах як один з найважливіших факторів функціонування і розвитку промислових фірм. Ця діяльність постійно вдосконалюється відповідно до об'єктивними вимогами виробництва та реалізації товарів, ускладненням господарських зв'язків, підвищенням ролі споживача у формуванні техніко-економічних та інших параметрів продукції.

У цьому ряду особливе місце займають комп'ютери та інше електронне обладнання, пов'язане з їх використанням в якості інструменту для діловодства і раціоналізації управлінської праці. Їх використання в якості інформаційних машин, що, хоча і не відповідає їх назві, але останнім часом стає основним видом їх застосування, дозволяє скоротити час, необхідний на підготовку конкретних маркетингових і виробничих проектів.

Об'єкт досліджень: мережа аптек.

Предмет досліджень: автоматизована інформаційна системи обліку товару мережі аптек.

Мета дослідження: розробка програми автоматизованої інформаційної системи обліку товару мережі аптек.



## РОЗДІЛ 1 АНАЛІТИЧНИЙ ОГЛЯД

Предметом дипломного проекту є діяльність мережі аптек.

Основною метою діяльності мережі аптек є забезпечення населення, закладів охорони здоров'я та інших установ, підприємств, організацій продукцією медико-фармацевтичного призначення.

На сьогоднішній день у світі інформаційні технології досягли високого рівня розвитку. У зв'язку з цим деякі мережі аптек використовують автоматичні засоби, що дозволяють ефективно зберігати, обробляти і розподіляти накопичені дані. Виходячи з сучасних вимог, що пред'являються до якості роботи підприємства, не можна не відзначити, що їх ефективна робота цілком залежить від рівня оснащення інформаційними засобами на базі комп'ютерних систем автоматизованого обліку.

При організації обліку в мережі аптек важливу роль відіграють програмні та технічні засоби автоматизації як мобільні, так і стаціонарні: сканери штрих кодів, принтери етикеток, термінали збору даних. Все це, разом з програмним забезпеченням, дозволяє вести облік товару в той момент, коли він переміщається або продається. Темп сучасного бізнесу не дозволяє відкладати процес реєстрації руху товарів.

У цьому зв'язку тема цього проекту є на сьогоднішній день актуальною значущою для ефективного функціонування мережі аптек

Мета проекту - розробити інформаційну систему обліку лікарських засобів, яка забезпечить коректне ведення даних.

Основним завданням даного проекту є створення діалогової автоматизованої системи управління торговельною діяльністю, що дозволяє усунути недоліки справжніх торговельних процесів.

Важливим завданням створення системи є скорочення часу на проведення основних операцій персоналом організації, що, безумовно, повинно спричинити за собою якісні покращення в діяльності організації.

Формування достовірної інформації у вигляді різних звітів і діаграм, для об'єктивної оцінки поточного стану справ на підприємстві.

Система повинна забезпечувати виконання функцій, пов'язаних з веденням обліку закупівель і продажу продукції, а саме програма повинна забезпечувати:

1. Коректне ведення даних (додавання, видалення, редагування записів);

2. Вести облік приходу/витрати товарів;

Загальною вимогою системи є створення дружнього інтерфейсу, за допомогою якого користувач міг би легко і швидко занести інформацію в базу даних, за перерахованим вище вимогам.

На додаток до основних функцій для практичного використання повинен бути розроблений користувальницький інтерфейс (зроблений компактніше і систематизований).

Програма повинна:

володіти розвиненим графічним інтерфейсом. З даною програмою зможуть працювати користувачі різної кваліфікації.

дозволяти заносити і зберігати інформацію в базі даних, редагувати її, формувати звіти.

значно зменшити витрати часу на складання поточної документації та підсумкових звітів, виключає можливість помилок у розрахунках, може використовуватися в різних масштабах, тобто може працювати з величезною кількістю даних, не вимагає перепрограмування при зміні моделей устаткування, характеристиках, ціни і т.п.

## 1.1 Необхідність автоматизації

Аптека - це в першу чергу точка, де здійснюються продажі, тому багато бізнес-процеси тут ті ж, що і в звичайному магазині. Перелік завдань, що підлягають автоматизації в окремій (самостійної) аптеці, включає в себе:

1. бухгалтерський облік;
2. замовлення товару у постачальників (широкий асортимент і наявність великого числа постачальників диктують необхідність реалізації електронного замовлення із застосуванням спеціальних можливостей для порівняння цін різних постачальників);
3. аналіз руху товару і формування аналітичних звітів по різних групах товарів;
4. реєстрація та облік продажів.

Для аптечної мережі список таких завдань буде більш довгим, в нього входять, наприклад, ведення класифікаторів товарів, централізований фінансовий і планово-економічний облік, логістика (перекидання товару зі складу в роздрібні точки), управління персоналом. Крім цього, залежно від прийнятих схем роботи замовлення товару у постачальників може здійснюватися або централізовано, або самими точками продажу (т.б. децентралізовано). Те ж відноситься і до ціноутворення: розцінка фарм-препаратів може здійснюватися як безпосередньо в аптеках, так і в центральному офісі.

Крім того, сьогодні в роздробі для активізації продажів застосовують цілий спектр інструментів для підвищення лояльності споживачів. Один з них - дисконтні програми різного типу. Вони включають в себе проведення обмежених за часом дії рекламних акцій; крім того, затребуване ведення автоматизованої роботи з лікарями і можливість бронювання товару по телефону. Інтегрований в систему довідник Відаль дозволяє безпосередньо з терміналу програми здійснювати вибір препарату з хвороб, АТХ і фармгруп, надаючи доступ до анотацій препаратів, інформації про аналоги і їх наявності в аптеці .

Однак крім завдань, схожих з завданнями будь роздрібної точки (або мережі магазинів), в аптечній справі існує чимало специфічних бізнес-процесів, що вимагають нестандартного підходу та інших (ніж у звичайній торгівлі) алгоритмів рішення.

По-перше, аптечні мережі працюють з величезним асортиментом і практично всі товари в аптеці мають терміни придатності. Це пред'являє специфічні вимоги до інформаційної системи, яка повинна розташовувати функціоналом для обліку термінів придатності декількох тисяч позицій. Ще однією особливістю товарного асортименту в аптеці є наявність декількох видів одного і того ж препарату: це може бути один медикамент в різних формах випуску - спрей, гель, мазь і т. Д., В різних дозуваннях, препарати для дітей, дорослих і т. д. ІС повинна дозволяти переглядати і вибирати товарні позиції, переміщаючись по каталогу товарів, створюючи підгрупи в межах однієї групи товару, і допомагати здійснювати оперативний пошук препарату в процесі роботи з клієнтом .

По-друге, в аптеках ведеться суворий партійний облік (іноді облік за серійними номерами) відповідно до вимог GMP, особливо якщо є власне виробництво (приготування ліків на замовлення). В останньому випадку даний стандарт (GMP) передбачає забезпечення певного документообігу, правил зберігання, а також облік по складу речовини в сировину. Тут є своя специфіка: всі вступники партії субстанцій, препаратів проходять в аптеках обов'язковий вхідний контроль. Всі дії, пов'язані з отриманням, виробництвом, розподілом, в тому числі і з контролем, описуються стандартними процедурами. Контроль виконується відповідно до фармакопеї, збіркою стандартів і положень, що регламентують якість лікарських засобів (ЛЗ). Унікальний номер присвоюється кожній партії для того, щоб можна було на готовий препарат зібрати досьє серії, де зазначаються місце та дата придбання усіх компонентів (субстанцій, допоміжних речовин), які входять в готовий препарат згідно специфікації, а також наводяться результати випробувань цих компонентів, умови виробництва і хід технологічного процесу (скільки, при якій температурі видерживалась суміш, які відбувалися реакції і т. д.). Крім того, існує таке

поняття, як специфікація. Це перелік речовин, що входять в готовий препарат. Зазвичай тут вказується, яку вологість і/або яку концентрацію активної речовини в субстанції повинні мати компоненти, з яких готується ЛЗ, а отже, і інформаційна система повинна мати функціонал для роботи з урахуванням маси, вологості, вмісту активної речовини .

У відношенні готових препаратів важливість партійного обліку також не викликає сумнів...

у, так як можуть виникнути ситуації, коли необхідно вилучити цілу партію бракованого товару. Виробникові треба знати, куди вона розійшлася - за якими торговим точкам, через якихось дистриб'юторів. ІС повинна давати можливість відстежити шлях руху бракованого товару практично до купівельного чека.

Експерти називають ще ряд функцій, обов'язкових для аптечної інформаційної системи: розцінка товару відповідно до правил регульованого ціноутворення на окремі групи ЛЗ, по серійний облік ЛЗ і відстеження забракованих серій, наявність функції відпустки за пільговими та безоплатними рецептами та персоніфікований облік відпустки даного виду ЛЗ хворим, які мають право на пільги; відстеження фальсифікатів, контроль за обов'язковим асортиментом, спеціальний облік ряду груп ЛЗ - сильнодіючі ЛЗ, групи А і Б та ін. Організація довідкової служби з функцією надання інформації про наявність товару в будь-якому підрозділі аптечній мережі в режимі реального часу, взаємодія автоматизованої системи управління аптекою з зовнішніми довідковими системами, робота з банківськими картами; можливість використання подарункових сертифікатів, промо акцій, торгівлі за зразками .

Для забезпечення всіх цих запитів необхідні ІС, що володіють відповідним функціоналом.

## 1.2 Які інформаційні системи потрібні аптеці

Базовий комплект ПЗ для аптеки повинен забезпечувати електронну взаємодію з постачальниками (від замовлення до приходування накладних),

ведення партійного обліку, контроль термінів придатності і забракованих серій медичних препаратів, можливість повернення товару постачальнику (наприклад, у випадках коли у ліки, що лежить в аптеці, закінчується термін придатності або виявляються якісь дефекти) і повернення товару від споживача (якщо, скажімо, виявляється, що дана партія - фальсифікат) переоцінку товару, межаптечну перекидання товару для мереж і аптек з віддаленими кіосками, роздрібну реалізацію (касовий модуль), інвентаризацію, вивантаження даних в систему бухгалтерського обліку та надання оперативної звітності.

Середня аптека сьогодні включає наступне обладнання: два-три автоматизованих торгових місця на базі фіскального реєстратора (комп'ютер, власне фіскальний реєстратор, сканер штрих кодів, дисплей покупця, грошовий ящик), один комп'ютер матеріально-відповідальної особи, один комп'ютер як сервер БД.

В аптеках з відкритою викладкою можуть також застосовуватися системи безпеки (зокрема, антикражні пристрої) і відеоспостереження. У великих торгових мережах вже існує практика інтеграції таких систем з ІС (наприклад, з касовими системами - для запобігання або припинення зловживань персоналу і покупців), але в аптечній справі, навіть у великих мережах це поки мало затребувано у зв'язку зі вторинністю такого завдання по відношенню до автоматизації основних бізнес-процесів.

Система контролю доступу (СКД) може застосовуватися для обліку відвідувань складу з медикаментами. Пов'язано це з тим, що аптеки працюють з наркотичними препаратами, і для запобігання зловживань з цими продуктами СКД дуже ефективні. Їх інтеграція з бек-офісом дозволяє створити єдиний інформаційний простір і в режимі реального часу контролювати відвідуваність складу. У разі будь-яких порушень система допомагає легко виявити винного. Крім того, це дає можливість жорстко обмежити і доступ до зберігається тут інформації. Таким способом топ-менеджер компанії отримує повне уявлення про систему безпеки у всій мережі і може вчасно відслідковувати виникнення нештатних ситуацій і вживати заходів для їх вирішення .

Ключовими критеріями при виборі ІС є:

- вартість базового комплексу і послуг технічної підтримки (що особливо критично для не мережевих аптек);
- можливість своєчасних доробок системи під нові вимоги ринку та побажання замовника (користувача);
- невисокі вимоги до обладнання (сервер, робоче місце директора, робоче місце касира) - це дозволяє уникнути додаткових витрат на закупівлю нового обладнання і використовувати вже наявні комп'ютери;
- підтримка системою великого спектру периферії (фіскальні реєстратори, сканери штрих коду, дисплеї покупця, термопринтери друку штрих коду і т. п.);
- наявність і доступна ціна додаткових модулів для збільшення функціоналу базового комплексу (особливо важливо для динамічно розвиваються аптечних установ);
- для використовуваної СУБД важливі вартість самої СУБД і додаткових ліцензій на робоче місце, кваліфікація ІТ-персоналу для адміністрування;
- здатність взаємодіяти з найбільш поширеними програмами бухгалтерського обліку.

Для аптечних мереж, особливо великих і динамічно розвиваються, ієрархія ключових критеріїв буде дещо інший, ніж для окремих аптек. Так, тут на перше місце вийдуть високі вимоги до СУБД, використовуваної в офісі для консолідації даних, а також до апаратної частини офісу (сервер, робоче місце менеджера). Для великих аптечних мереж рекомендується обладнання, яке гарантуватиме відмовостійкість і безперебійність в роботі, так як збої в офісному обладнанні можуть чинити негативний вплив на всю аптечну мережу.

Дуже важливо, щоб в ІС була можливість оперативної консолідації даних, для швидкого обміну інформацією про наявність товару в режимі реального часу як між декількома аптеками однієї мережі, так і між аптеками мережі і центральним офісом. Центральний офіс отримує від аптек звіти про реалізацію товарів і на їх основі здійснює планування закупівель.

Надзвичайно важлива також масштабованість ІВ, особливо на сучасному етапі, коли аптечні мережі швидко збільшуються. Якщо компанія росте більш ніж на 30% на рік, це означає, що система повинна легко справлятися зі збільшенням числа користувачів.

Велика кількість периферійного обладнання, підтримуваного системою, набуває особливої актуальності для міжрегіональних аптечних мереж. Сьогодні не можна забувати також і про еквайрингових системах: забезпечення роботи з кредитними картами стало вже поширеною практикою, а відхід від окремого EFT-терміналу до вбудованого знижує час обслуговування клієнта і, головне, усуває помилки, викликані додатковим ручним введенням інформації.

Важливий для сетих аптек і такий критерій, як вартість доопрацювання ПО за індивідуальними замовленнями: як правило, великі аптечні мережі з часом виробляють власне ноу-хау, слідуючи маркетинговим тенденціям у своїх регіонах, і надають запити на доопрацювання ПЗ з урахуванням місцевої специфіки.

Іншими словами, необхідно звернути увагу на наступне: клієнт зрештою купує не програма, а технологію, в якій істотні не тільки функціонал ІС, але і час впровадження, вартість проекту і сукупна вартість володіння продуктом (ТСО , що включає в тому числі і вартість адаптації та супроводу).

### 1.3 Тенденції та перспективи

Як уже згадувалося, однією з найбільш серйозних тенденцій сьогодні є інтенсивний розвиток аптечних мереж та зменшення кількості незалежних аптек. Зрозуміло, що не тільки повноцінний розвиток, але і саме існування торговельної мережі нині неможливе без використання відповідних ІТ. Саме тому, найбільш затребуваними будуть рішення, що дозволяють централізовано, але в той же час гнучко управляти асортиментом, маркетинговими програмами, ціновою політикою в рамках аптечної мережі.



При цьому повинні дотримуватися високі вимоги до надійності роботи ПЗ в аптеках з тим, щоб не росли витрати, пов'язані з його експлуатацією.

Один з нових напрямків на цьому ринку, створення компаній, що беруть на аутсорсинг управління товарними запасами аптеки, ціновою політикою, маркетинговими програмами. Аптеки при цьому зберігають юридичну та фінансову незалежність і отримують наявні у аптечних мережах переваги відносно рівня цін (при закупівлях товару у постачальників) і використання праці висококваліфікованих (і, отже, високооплачуваних) менеджерів.

Цікавою тенденцією на фармацевтичному ринку є і народження нових торговельних форматів: так, сьогодні актуальні вже не просто аптеки, де продаються тільки лікарські препарати, але так звані магазини здоров'я *gaquo* ;, де можна купити препарати по догляду за зовнішністю, отримати консультацію офтальмолога, який може працювати тут же, і т. п. Для забезпечення роботи таких комплексних торгових центрів, потрібні ІС з розширеним спектром можливостей.

Однак і незалежні аптеки не бажають здавати позиції, прагнучи до оптимізації роботи, до прозорості обліку руху товару і фінансових коштів за допомогою ІТ. Адже невеликим аптекам найчастіше досить автоматизації приходования товару, друку накладних та цінників і ведення необхідної первинної документації. Для самостійних аптек (втім, і для багатьох мереж) важливими критеріями є зручність системи в експлуатації, простота в навчанні персоналу роботі з ПЗ, а також вартість і якість обслуговування.

Сьогоднішній стан автоматизації аптечного роздробу в чому нагадує ситуацію із впровадженням ІТ в торгових мережах кілька років тому, коли комплексний підхід до автоматизації тут ще тільки зароджувався, не всі види ІС були затребувані, не було особливих вимог до інтеграції систем. До речі, і той факт, що в бесідах з експертами нам не вдалося отримати оцінки загального обсягу цього сегмента ринку, говорить про початковий...

стадії його розвитку. " На даному етапі автоматизації аптечних закладів комплексний підхід в основному полягає у впровадженні бухгалтерської та облікової системи. Одне лише скорочення дублювання введення вивільняє до

50% часу персоналу, як правило, має високу кваліфікацію і одержує за свою працю чималу винагороду. Крім того, надання доступу до даних за всю історію роботи аптечного підприємства дає можливість використання спеціалізованих систем прогнозування попиту.

Сьогодні в секторі традиційного (не фармацевтичного) ритейлу аналітичні системи і пакети для планування і прогнозування вже використовуються найбільш просунутими учасниками ринку, і недалекий той день, коли вони стануть невід'ємною приналежністю кожної поважаючої себе торговельній мережі. Тому той факт, що про подібні системах вже замислюються учасники фармацевтичного ринку, ясно говорить про те, що він активно набирає обертів. І цілком можливо, що темпи його розвитку будуть вищі, ніж свого часу у звичайного (не фармацевтичного) ритейлу; спираючись на досвід останнього (у поєднанні з прийшли сюди вже помітними капіталами), сегмент аптечного роздробу може зробити найближчим часом дуже істотний ривок.

#### 1.4 Аналіз ринку програмних продуктів

в даний час, працювати в аптеці без програми обліку неможливо, виникає наступне питання яка з програм краще, і яка найбільш підходить для роботи в аптеці.

-Парацельс- Це програмне забезпечення є спеціалізованим і розробленим виключно для аптек. Перша версія програми з'явилася в 1997 році. З тих пір з кожним оновленої версії тільки нарощувала свою функціональність. На сьогоднішній день програмний комплекс «Парацельс» (до 2013 року носив назву «Медбрат») п'ятої версії - це потужний і надійний спосіб для ведення аптечного бізнесу як одиночної аптеки так і аптечної мережі.

#### «Плюси»

- Невелика вартість програми, в порівнянні з аналогічними програмами, можливість безвідсоткового відстрочки;

- Готова база найменувань (86тис. Позицій), контроль націнки на товари, які контролюються державою.

- База аналогів і інструкції (інструкції) до препаратів;

- Внесення прибуткової документа (накладної) в ручному або автоматичному режимі - синхронізація позицій згідно кодів «Моріон» - що дає можливість автоматично, з першої накладної, оприбуткувати і розцінити товар;

- Інвентаризація без зупинки продаж, унікальна функція програми «Парацельс»;

- Програма для аналізу цін постачальників і формування замовлення "Robodok", автоматичне формування дефектури (ABC-аналіз, заявка працівників першого столу ...), Додатковою можливістю є автоматичне замовлення - за допомогою якого програма автоматично вибирає позиції за найдешевшими цінами на ринку, згідно складлася дефектури;

- Можливість реалізації препаратів за допомогою сканера - по заводським або внутрішніх штрих кодах;

- Реалізація медикаментів і виробів медичного призначення іншим юридичним особам, при реалізації державним установам програма контролює націнку передбачену для держ. Підприємств, також є можливість формувати видаткову накладну в форматі ММО і EXP;

«Мінуси»

- Як таких «мінусів» в програмі «Парацельс» НІ, невеликим «мінусом» можна назвати відсутність бухгалтерських операцій (вони також відсутні в програмах «Аптека» і «Скарб»), але у всіх цих програмах є можливість синхронізації з «1С Бухгалтерія» .

-1С (1С-Аптека)- одна з найпоширеніших програм в Україні, але тут основне питання, 1С, спочатку, розроблялася як програма для ведення бухгалтерії та обліку підприємств, але робота в Аптеці дуже сильно відрізняється від будь-якої іншої галузі. Саме тому не є найкращою для ведення обліку саме в аптеці.

«Плюси»

- Ведення бухгалтерії - так як 1С, спочатку, розроблялася як бухгалтерська програма, як висновок в ній передбачені всі основні бухгалтерські операції;

- Гнучкість - можливість індивідуального доопрацювання програми, тут не потрібно забувати про те, що робота програміста 1С достатньо дорога і кожна додаткова функція в програмі обійдеться вам в додаткові витрати.

#### «Мінуси»

- Відсутність готової бази найменувань, є однією з основних причин, відмови від 1С і перевагу іншій програмі для аптек, так як в фармації використовується контроль націнки на ряд препаратів, і сам перелік лікарських засобів, БАДів і виробів медичного призначення досить великий це від 20 до 150 тис найменувань

- Аналіз цін постачальників і замовлення товару - ця функція просто відсутня в базовій комплектації 1С-Аптека.

- Автоматичне завантаження і розцінка накладних - тут ми маємо на увазі, що при першому завантаженні накладних, потрібно «підбирати збігом» (підв'язувати) кожен з позицій, кожного постачальника. Також передбачений контроль націнки на препарати, які контролюються законодавством.

- Програмний комплекс «Аптека» (Моріон)- програма розроблена компанією «Моріон», як тільки аптечна програма, і має всі необхідні функції для роботи в аптеці, але при такій яскравій, початкової, характеристиці програма «Аптека» також має свої недоліки.

#### «Плюси»

- Готова база найменувань (150 тис позицій)

- Прив'язка до сайту «Моріон», що дає можливість, як приклад, контролювати ціни продажу медикаментів державним установам, але на практиці тільки поодинокі аптеки беруть участь в тендерах держ. закупівлі;

- База аналогів;

- Інструкції (анотації) препаратів;

#### «Мінуси»

- Основним і найвагомим «мінусом» програми «Аптека» досить велика помісячна, орендна плата за програму, яка підв'язана до курсу долара в Україні, тим самим ви збільшуєте поточні витрати аптеки (все решта програми - продаються)

- Складність приймання товару «дрібних» постачальників - мається на увазі, що в кожній аптеці, крім медикаментів достатньо великий асортимент, БАДів, косметики, дитячого харчування. . . Цей товар поставляють, так звані «дрібні» постачальники, які мають можливість скидати електронні накладні, але в цих накладних форумі коду «Моріон» і це унеможлиблює автоматичну обробку програмою «Аптека», виходом є внесення товару «вручну», тобто вносити кожну позицію, серію, ціну і т.д. , Що призводить до значної втрати часу. (Найкраще ця функція втілена в програмі «Парацельс» - в якій є можливість вибирати один з двох варіантів оприбуткування накладних (з кодом або без коду «Моріон»).

- Скарб- як «Парацельс» і Програмний комплекс «Аптека», Скарб розроблявся виключно для аптек, також має набір всіх необхідних функцій для роботи в аптеці.

«Плюси»

- Готова база найменувань (23 тис позицій)

- Внесення прибуткової документа вручну або автоматично - є можливість вносити накладні в форматі xls.

«Мінуси»

- Відносно невелика база найменувань

- Обов'язкова оплата технічної підтримки;

- Відсутність кодів «Моріон», що ускладнює внесення накладних.

## РОЗДІЛ 2 ІНФОРМАЦІЙНА МОДЕЛЬ

### 2.1 Інформаційна модель об'єкту

інформаційна модель- модель об'єкта, представлена у вигляді інформації, яка описує суттєві для даного розгляду параметри і змінні величини об'єкта, зв'язку між ними, входи і виходи об'єкта і дозволяє шляхом подачі на модель інформації про зміни вхідних величин моделювати можливі стану об'єкта. Інформаційні моделі не можна помацати або побачити, вони не мають матеріального втілення, тому що будуються тільки на інформації. Інформаційна модель - сукупність інформації, що характеризує суттєві властивості і стану об'єкта, процесу, явища, а також взаємозв'язок із зовнішнім світом.

Інформаційні моделі діляться на описові і формальні.

Описові інформаційні моделі - це моделі, створені на природній мові (тобто на будь-якій мові спілкування між людьми: англійською, російською, китайською, мальтійському і т.п.) в усній або письмовій формі.

Формальні інформаційні моделі - це моделі, створені на формальній мові (тобто науковому, професійному або спеціалізованому).

Приклади формальних моделей: всі види формул, таблиці, графи, карти, схеми і т.д.

Хроматичні (інформаційні) моделі - це моделі, створені на природній мові семантики колірних концептів і їх онтологічних предикатів (тобто на мові смислів і значень колірних канонів, репрезентативно відтворені у світовій культурі). Приклади хроматичних моделей: "атомарна" модель інтелекту (АМІ), міжконфесійна іманентність релігій (МИР), модель аксіолого-соціальної семантики (МАС) і ін., Створені на базі теорії та методології хроматизму.

Різноманіття проблемних ситуацій, що виникають в економіці та управлінні, диктує необхідність володіння технологіями моделювання процесів управління економікою. В силу того, що управління підлягають складні економічні об'єкти, для їх вивчення створюються моделі - "копії" досліджуваних реальних об'єктів. Моделі повинні бути точні і придатні для вивчення та практичного застосування. Це означає, що модель повинна мати достатній ступінь відповідності об'єкта моделювання для параметрів, які істотно впливають на результат. Ті ж параметри, які не впливають на результат, можна виключити, досягаючи необхідного відповідності модельованого об'єкту.

Типи інформаційних моделей

Виділяють кілька типів інформаційних моделей, що відрізняються за характером запитів до них:

- Моделювання відгуку системи на зовнішній вплив.
- Класифікація внутрішніх станів системи.
- Прогноз динаміки зміни системи.
- Оцінка повноти опису системи і порівняльна інформаційна значимість параметрів системи.

- Оптимізація параметрів системи по відношенню до заданої функції цінності.
- Адаптивне управління системою.

Рівні моделей:

- структури;
- поведінки;
- результатів.

Характеристики моделей:

- Мета (target).
- Точка зору (view).
- Повнота.
- Цілісність і несуперечливість.
- Адекватність і узгодженість з оригіналом (coherent).
- Складність.
- Надмірність.
- Архітектура.

Суть проблеми прийняття рішень із застосуванням моделей полягає в використанні інформаційних технологій, що забезпечують найкращий результат, оцінюваний критерієм оптимальності. Проблемні ситуації в управлінні, як правило, мають багатоальтернативного характер. Для оптимального вирішення такої ситуації тільки особистого досвіду недостатньо, необхідний комплекс взаємопов'язаних моделей.

Формалізація поставленого завдання управління дозволяє чітко усвідомити суть проблемної ситуації і отримати оптимальне рішення. Якщо формалізована задача занадто важка, то її можна розбити на частини, знизивши складність підзадач і використовуваних для їх вирішення моделей.

Кожен фрагмент завдання управління вирішується узгодженими математичними методами за допомогою відповідного алгоритму. Об'єднання моделей в математичні пакети відкриває великі можливості для експериментування. Змінюючи вхідні параметри пакетів моделей, можна швидко отримати кінцевий результат, навіть не вникаючи особливо алгоритму.

Інформаційна технологія включає в себе всі етапи вирішення проблемної ситуації, починаючи від постановки задачі управління, конструювання моделі та закінчуючи чисельним рішенням, результати якого впливають, в кінцевому рахунку, на характер прийнятого рішення.

Виділяють вісім технологічних етапів моделювання процесів управління економікою.

Етап 1. Аналіз особливостей об'єкта моделювання. Прикладом можуть бути склад готової продукції, організація випуску нової продукції, системи транспортних перевезень і т.п.

Етап 2. Аналіз проблемної ситуації, що склалася в даному об'єкті моделювання. Наприклад, для нормальної організації виробництва існує ряд альтернатив і необхідно вибрати альтернативу функціонування складу готової продукції, виявити залежність між темпом споживання продукції, часом поставки, розмірами складських площ і обіговими коштами, які завжди обмежені.

Етап 3. Визначення неспостережуваних параметрів, що дозволить усвідомити цілі моделювання і спростити формалізацію задачі управління.

Етап 4. Визначення спостережуваних параметрів, що обґрунтовує постановку задачі і допомагає знайти її оптимальне рішення, з огляду на обмеження, які накладаються на шукані змінні. Наприклад, при визначенні розмірів оптимальної партії поставки взуття в магазин враховується, що отримане рішення повинно бути цілочисельним і т.п.

Етап 5. Параметри адекватності, які встановлюються в залежності від необхідного результату. Наприклад, якщо головне призначення туфель - захист від холоду і дощу, то важливі параметри міцності, зручності і непромокальності. Якщо туфлі - предмет вечірнього туалету, то головним параметром адекватності буде привабливість дизайнерського рішення.

Етап 6. Використовуваний математичний апарат, який вибирається в залежності від особливостей поставленого завдання.

Етап 7. Аналіз отриманих результатів моделювання, який проводиться на основі

отриманих чисельних оцінок дозволеної проблемної ситуації.

Етап 8. Прийняття рішення. Ухвалення рішення - неформальний процес, який цілком залежить від досвідченості керівника і може частково або повністю не збігатися з отриманими в результаті моделювання чисельними оцінками. Проте ухвалення рішення значно прояснюється, якщо проблемна ситуація формалізована, і отримані результати моделювання відображають стійкі тенденції в розвитку економічної проблемної ситуації.

Особливо слід відзначити неформальні етапи, де визначальну роль мають кваліфікація і досвід роботи сучасного менеджера-економіста:

- ретельний аналіз проблемної ситуації, що виникає в економіці, необхідний для адекватного опису (постановка завдання);
- аналіз результатів моделювання і висновки;
- залежність ефективності прийнятого рішення від правильності постановки завдання і висновків, зроблених на етапі 7.

У зв'язку з цим менеджер повинен заздалегідь забезпечити збір і обробку повноцінної інформації, що допомагає йому оцінити оптимальність прийнятого рішення.

Щоб отримати сукупності інформаційних ресурсів, найбільш повно пояснюють отримані результати і дають підставу для їх прогнозування, використовуються три види інформаційних технологій моделювання: натурного, експертного та математичного.

## 2.2 Інформаційна система діяльності мережі аптек

Інформаційна система мережі аптек пов'язують користувачів або програмні комплекси, забезпечуючи їх взаємодія завдяки зрозумілому машинам формалізованого мови і опосередкованим інструментів спілкування. Вони дозволяють накопичувати інформацію або передавати її в заданому форматі споживачеві (людині або програмою) в режимі реального часу або з відстрочкою. Метою інформаційних систем є, в першу чергу, допомога фахівцям в ухваленні рішення.

Істотну роль при цьому відіграють різні інформаційні моделі. Сам термін "інформаційна модель" має різне тлумачення у різних авторів і застосовувався ще до появи комп'ютерів. В даному випадку під ІМ розуміється структура (система), що включає в себе якісь інформаційні об'єкти, зв'язку і правила взаємодії, що описуються на спеціалізованій мові.

Іноді інформаційний об'єкт представляють у вигляді метаданих. У найпростішому випадку вони організовані у вигляді регулярно оновлюваного словника, який класифікує і співвідносить продукти, технології та бізнес-концепції, відображаючи всі зміни бізнесу компанії. Інформаційна модель реалізується у вигляді програмного комплексу і має інтерфейс для взаємодії з користувачем або іншим програмним комплексом.



Прикладом інформаційних моделей можуть служити такі програмні комплекси, як "Кадри", АСБУХ (автоматизована система бухгалтерського обліку), ERP- системи планування ресурсів підприємства, CRM-управління клієнтськими відносинами та інші.

Основу функціонування системи управління мережі аптек складає управлінська діяльність органів управління, яка здійснюється посадовими особами і включає рішення пов'язаних між собою задач управління. Окремі завдання управління і управлінська діяльність в цілому являють собою процес переробки інформації. Переробку інформації здійснюють посадові особи органів управління шляхом виконання інформаційних процедур.

Управлінську діяльність органів управління та посадових осіб, завдання управління, а також інформаційні процедури в подальшому ми будемо називати елементами управлінської діяльності, здійснюваної системою управління.

Питання автоматизації професійної діяльності є в даний час досить актуальними і не до кінця опрацьованими. В даному пункті ці питання будуть розглянуті стосовно до проблеми автоматизації управлінської діяльності як найбільш складної для будь-якого фахівця в області економіки, т. Е. Описані типи об'єктів автоматизації в системі управління персоналом, а також характеристики відомих підходів до автоматизації управлінської діяльності та порядок проведення інформаційного обстеження управлінської діяльності.

### 2.3 Об'єкт автоматизації в системі організації мережі аптек

Розглядаючи питання автоматизації систем управління, перш за все, необхідно чітко визначити, що ми збираємося автоматизувати, т. б. визначити об'єкти автоматизації.

Для визначення об'єктів автоматизації в системі управління персоналом необхідно проаналізувати процес її функціонування, її склад та оцінити потреби. В результаті аналізу повинно бути отримано опис процесу

переробки інформації в системі управління, визначені елементи цього процесу і зв'язку між ними.

Під системою управління персоналом розуміється сукупність функціонально пов'язаних органів управління, пунктів управління, систем зв'язку, систем і засобів автоматизації управління, а також спеціальних систем, що забезпечують збір, обробку та передачу інформації.

Основу системи управління складають органи управління, які виробляють керуючі впливи (накази, директиви) і тим самим здійснюють управління. До складу органів управління включаються: правління, дирекція, адміністрація управління, відділи та інші органи. Органи управління розміщуються на пунктах управління та, використовуючи системи зв'язку, системи та засоби автоматизації управління, а також спеціальні системи, виконують свої функції.

У процесі виконання своїх функцій органи управління здійснюють управлінську діяльність, яку можна розглядати як процес переробки інформації. Дійсно, орган управління отримує вхідну інформацію по різних каналах (накази, директиви вищих органів управління, донесення нижчестоящих органів управління, а також іншу інформацію). Далі він аналізує отриману інформацію, певним чином перетворює її і створює нову інформацію, яку по каналах зв'язку передає в підлеглі і вищі органи управління (накази, розпорядження підлеглим, донесення вищих органів управління).

Управлінська діяльність може бути представлена як сукупність певним чином пов'язаних завдань управління. Кількість, складність завдань управління, що вирішуються в процесі управлінської діяльності, а також вимоги щодо оперативності їх вирішення можуть бути різними в різних умовах обстановки. Однак перелік і зміст завдань управління, як правило, є стабільними - незмінними або слабо змінними - в процесі життєдіяльності організації на досить великому інтервалі часу. Результатами виконання завдання управління є керуючі впливи на підлеглих і надання необхідної інформації вищестоящим органам управління, які оформляються у вигляді документів (наказів, директив, розпоряджень, а також донесень, звітів і

довідок).

Завдання управління вирішуються посадовими особами органів управління та так же, як управлінська діяльність, є процес переробки інформації.

аналіз процесу переробки інформації при вирішенні будь-якої задачі управління дозволяє виділити в ньому три типи взаємопов'язаних інформаційних процедур, які полягають в реалізації того чи іншого механізму переробки вхідної інформації в конкретний результат і індивідуально виконуваних посадовими особами.

1. Повністю формалізуються інформаційні процедури, при виконанні яких алгоритм переробки інформації залишається незмінним і повністю визначений. До таких процедур відносяться пошук, облік, зберігання, передача інформації, друк документів, розрахунок заробітної плати, підведення підсумків діяльності підприємств і фірм, розрахунок на моделі показників ефективності діяльності підприємств і фірм і т. Д. Повністю формалізуються процедури найкраще піддаються автоматизації із застосуванням ЕОМ. Вони можуть виконуватися без участі або з мінімальною участю людини і не вимагають високого рівня його підготовки.

2. Формалізації інформаційні процедури, при виконанні яких створюється нова унікальна інформація, причому алгоритм переробки вихідної інформації невідомий. Принципово існують дві процедури такого типу: формування безлічі альтернатив вибору (наприклад, варіантів вибору інвестиційних проектів) і власне вибір одного варіанта з даної множини. Як правило, такі процедури реалізуються посадовими особами з використанням результатів виконання інформаційних процедур першого типу. Вимоги щодо знання процесів функціонування підприємств і фірм до осіб, які виконують формалізації інформаційні процедури, дуже високі.

3. Погано формалізуються інформаційні процедури, при виконанні яких алгоритм переробки інформації може змінюватися і повністю не визначений. До погано формалізованих процедур відносяться завдання планування, оцінювання ефективності варіантів побудови фінансової політики фірми і т. Д. Погано формалізуються процедури не можуть виконуватися без участі людини. До людини, що виконує погано формалізуються інформаційні процедури, пред'являються високі вимоги щодо знання процесів функціонування економічних систем та алгоритмів переробки інформації. Вони виконуються однією посадовою особою і включають, як правило, кілька повністю формалізуються і формалізації процедур, порядок проведення яких визначає посадову особу, виходячи з особливостей розв'язуваної задачі управління в кожному конкретному випадку.

Таким чином, основними типами інформаційних процедур, індивідуально виконуваних посадовими особами, є повністю формалізуються і формалізації процедури. Введення поняття погано формалізуемой процедури зручно як проміжне опис процесів переробки інформації, а також для виділення в цих процесах досить цілісних елементів, які виконуються одним або кількома посадовими особами (наприклад, оцінка ефективності проведеної фінансової операції, розробка варіанту плану виведення фірми з кризового стану і т. д.).

Необхідно відзначити, що поділ процесу переробки інформації на окремі інформаційні процедури не є абсолютним, раз і назавжди визначеним. Цей поділ відповідає ступеню вивченості об'єкта управління і системи управління. У міру вивчення об'єкта управління і отримання нової інформації про нього формалізації інформаційні процедури можуть бути спочатку замінені погано формалізуються, а в подальшому - і повністю формалізуються інформаційними процедурами. Саме така послідовна заміна інформаційних процедур, що виконуються в процесі розробки управлінських рішень, лежить в основі одного з підходів до автоматизації управлінської діяльності, про що буде сказано нижче.

аналіз функціонування системи управління персоналом дозволяє виділити ієрархічно пов'язані об'єкти автоматизації як елементи управлінської діяльності, здійснюваної системою управління.

1. Управлінська діяльність органів управління та посадових осіб.
2. Завдання управління персоналом, які вирішуються керівництвом фірми в цілому або посадовими особами (посадовою особою) в процесі управлінської діяльності.
3. Повністю і погано формалізуються інформаційні процедури, індивідуально виконуються посадовими особами (можливо, з використанням різних технічних пристроїв) при вирішенні різних завдань управління.

Рівень автоматизації управлінської діяльності, Завдань управління та інформаційних процедур може бути різним. Він залежить від можливостей по розробці програмного забезпечення, наявних технічних засобів, а також від інших причин, в тому числі і чисто психологічних, пов'язаних з готовністю і бажанням посадових осіб використовувати в своїй роботі ЕОМ. Однак кожному об'єкту автоматизації можна поставити у відповідність

типове програмне засіб, який доцільно застосовувати при автоматизації конкретного елемента процесу переробки інформації.

Інформаційні моделі діяльності мережі аптек включають перелік завдань управління, що вирішуються в процесі цієї діяльності, а також вихідну і вхідну інформацію, необхідну для вирішення відповідного завдання. Крім того, необхідно вказівку на підпорядкованість і відносну важливість завдань, що входять в інформаційну модель.

Відносна важливість завдань визначається з істоти і цілей управлінської діяльності. Підпорядкованість завдань проявляється в тому, що окремі результати вирішення одних завдань можуть бути вихідними даними для вирішення інших.

Подальша деталізація інформаційних моделей діяльності посадових осіб та органів управління здійснюється шляхом побудови інформаційних моделей входять в них завдань управління.

Інформаційні моделі задач управління формулюються як сукупності інформаційних процедур і зв'язків між ними. Інформаційні моделі задач управління, як правило, включають погано формалізуються процедури, які в подальшому повинні досліджуватися з метою побудови їх інформаційних моделей. Методика побудови інформаційних моделей задач управління включає наступні впорядковані етапи робіт.

Етап 1. Визначення вихідних даних. Вихідні дані повинні бути конкретизовані до параметрів, які доцільно об'єднати в групи. Кожна група містить параметри, тісно пов'язані між собою і визначаються спільно. Групи вихідних параметрів, як правило, визначаються послідовно, тому необхідно вказати послідовність формування груп цих параметрів. Групи параметрів повинні бути об'єднані в документи, які виробляються в процесі управління.

Послідовність визначення переліку вихідних даних може бути різною: спочатку формується перелік вихідних документів, а потім - перелік визначальних їх параметрів (груп параметрів), або навпаки.

Етап 2. Визначення інформаційних процедур, результатом яких є виділені групи вихідних параметрів об'єкта автоматизації.

Групи вихідних параметрів можуть визначатися усіма типами

інформаційних процедур - повністю формалізуються, що погано формалізуються і формалізації. При відсутності автоматизації групи вихідних параметрів визначаються, як правило, з використанням формалізації і найпростіших формалізованих процедур (розрахунків). У процесі інформаційного обстеження формалізації процедури можуть замінюватися на погано формалізуються, що призводить до істотної зміни діяльності посадових осіб в процесі управління.

Наприклад, в задачі планування фінансової діяльності підприємства одним з вихідних документів є план господарсько-фінансової діяльності, що містить групи параметрів, що визначають:

- аналіз обсягу виробництва і реалізації продукції;
- аналіз оборотного капіталу;
- факторний аналіз прибутку і т. д.

Розробка зазначеного плану починається з аналізу змін у складі та структурі активів і пасивів підприємства, далі аналізується обсяг і структура продукції, що випускається, проводиться аналіз собівартості продукції і витрат обігу і т. д.

Найбільш простий інформаційною моделлю є модель повністю формалізуемой інформаційної процедури, що включає вхідні та вихідні дані, а також алгоритм (механізм) переробки інформації. Оскільки за визначенням відомий алгоритм переробки інформації в рамках такої процедури, при побудові її інформаційної моделі принципових труднощів не виникає. Інформаційна модель повністю формалізуемой процедури не обов'язково повинна включати детальний алгоритм переробки інформації. Достатнім є його словесний опис (наприклад, вказівка на необхідність використання відомих прикладних математичних методів), з якого можна зробити висновок про можливість побудови такого алгоритму.

інформаційна модель формалізації інформаційної процедури включає вихідні параметри (то, що має виходити в результаті інтелектуальної діяльності посадових осіб), тип розв'язуваної задачі (формування безлічі варіантів або вибір), а також необхідні вихідні дані для вирішення завдання.

Основна складність створення моделі формалізації інформаційної

процедури полягає у визначенні необхідних для її виконання вихідних даних. Остаточне рішення про склад необхідних вихідних даних приймає посадова особа, діяльність якого автоматизується. У завдання дослідника входить з'ясування у посадової особи переліку необхідних для проведення формалізації процедури вихідних даних.

Крім того, за результатами аналізу процесу управління дослідник може запропонувати посадовій особі додаткові дані, які можуть бути отримані шляхом виконання повністю або погано формалізуються процедур і які можуть виявитися корисними при виконанні формалізації процедури. Якщо посадова особа погодиться з корисністю запропонованої додаткової інформації, в інформаційну модель задачі управління включається погано формалізується інформаційна процедура замість розглянутої спочатку формалізації інформаційної процедури.

Побудова інформаційної моделі погано формалізуємою інформаційної процедури означає практично адекватну заміну даної процедури сукупністю повністю формалізуються і формалізації інформаційних процедур. Оскільки погано формалізується інформаційна процедура при автоматизації діяльності посадових осіб розглядається як альтернатива існуючої формалізації процедури (найчастіше - процедури формування варіантів вирішення поставлених завдань), побудову її інформаційної моделі може викликати суттєві труднощі. Труднощі ці обумовлені необхідністю детального вивчення процесу бойових дій і управління ними.

На закінчення відзначимо, що в даний час отримали досить широке поширення так звані CASE - методології і технології, що дозволяють системно підійти до розробки програмного забезпечення АІС різного призначення на всіх етапах його життєвого циклу.

В даний час існують різноманітні види моделей підприємства, які використовуються для вирішення різних завдань: визначення середнього завантаження обладнання, потрібних виробничих потужностей, управління транспортом і т.д. Для цілей автоматизації використовуються типи моделей, найбільш повна класифікація яких міститься в стандартах IDEF (Integrated Computer Automated Manufacturing DEFinition).

Ідея створення IDEF зародилася в середині 70-х років у ВПС США як вирішення проблеми підвищення продуктивності і ефективності інформаційних технологій, яка виникла, в свою чергу, при реалізації програми ICAM (Integrated Computer Aided Manufacturing). IDEF розглядався як сімейство методів і технологій для створення моделей складних систем і проектування комп'ютерних систем. Всього передбачалося створити 14 стандартів, в їх числі:

IDEFO - Моделювання функцій

IDEF1 - Інформаційне моделювання

IDEF1X - Моделювання даних

IDEF2 - Динамічне моделювання

IDEF3 - Опис процесів

IDEF4 - Об'єктно-орієнтовані методи проектування

IDEF8 - Інтерфейс користувача

IDEF10 - Технічне проектування

IDEF14 - Проектування обчислювальних мереж.

Початок сімейства стандартів поклав стандарт IDEFO, розроблений на основі технології моделювання, відомої як SADT (Structured Analysis & Design Technique).

У IDEF1 був розширений і перейменований в IDEF1X. Частина стандартів IDEF так і не перетворилася на стандарти де-факто, наприклад стандарт IDEF2. Інша частина, наприклад, IDEFO і IDEF1X, перетворилася зі стандартів BBC в стандарт уряду США, відомий як FIPS. В цілому сімейство стандартів IDEF охоплює практично всі реально використовувані в даний час підходи до створення моделей підприємств, які застосовуються при автоматизації процесів, і методи проектування (структурний і об'єктно-орієнтований підходи). В області створення моделей підприємства стандарти IDEF виконують таку ж роль, як модель стека протоколів OSI в області створення мереж і додатків.

Для створення динамічних моделей процесів найбільшого поширення в даний час отримали методи, засновані на застосуванні мереж Петрі і кінцевих автоматів. Іноді для цих цілей використовуються моделюють



системи, в яких застосовуються динамічні елементи різної природи, наприклад, описувані системами звичайних нелінійних диференціальних рівнянь, елементи, які використовуються в теорії автоматичного регулювання (чисте запізнювання, порогові елементи) і т. д

Таким чином, аптечна мережа включає в себе:

- центральний офіс;
- периферійні точки (аптеки і аптечні пункти).

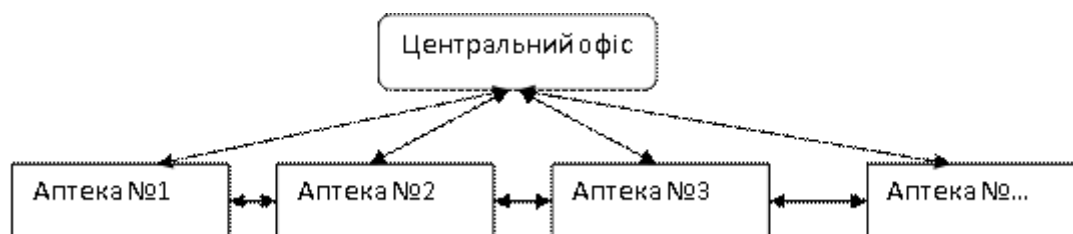


Рисунок 3.1.1 – Аптечна мережа

У центральному офісі зберігається зведена інформація про товар (замовлення, парафії, продажу та т. Д.) з усіх аптек. На кожній аптеці зберігається вся довідкова інформація і документи, що відносяться тільки до даної аптеці. Основні довідники (медикаменти, постачальники) вводяться централізовано. Документи, що відображають рух товарів, можуть вводитися як в центрі і надходити на аптеку по каналах зв'язку (наприклад, парафії товарів), так і в аптеці і надходити в центр (наприклад, продажу). Ціноутворення може відбуватися як централізовано, так і незалежно на кожній аптеці. Повноваження на замовлення медикаментів також можуть розподілятися між центральним офісом і аптеками.

Основна мета автоматичної інформаційних систем - облік наявності товару в мережі аптек.

Менеджер може як вибрати вже існуючу позицію, так і створити нову товарну позицію.

Після імпорту або введення інформації менеджер звіряє кількість позицій і загальну суму документа. Якщо виявлені розбіжності в документі, менеджер вносить корективи.

Введена інформації разом з новими цінами реалізації автоматично повинна передається в аптеку по каналах зв'язку.

В результаті пошуку отримує список товарів. У списку для кожного відображається така інформація:

- одиниця виміру;
- ціна реалізації;
- номер аптеки мережі.

Якщо товар не вдалося визначити чи по даному товару відсутня інформацію, система виводить на екран відповідне повідомлення про помилку. В цьому випадку в системі фіксується факт відсутності товару в аптеці.

## РОЗДІЛ 3 ОПИС ПРОГРАМИ

### 3.1 Опис середовища програмування

В якості мови програмування був обраний Delphi XE7 на базі середовища Embarcadero RAD Studio XE7

Delphi - імперативний структурований об'єктно-орієнтована мова програмування з суворою статичною типізацією змінних. Основна область використання - написання прикладного програмного забезпечення.

Embarcadero Delphi, раніше Borland Delphi і CodeGear Delphi, - інтегроване середовище розробки ПО для Microsoft Windows, Mac OS, iOS і Android на мові Delphi (раніше мав назву Object Pascal), створена спочатку фірмою Borland і на даний момент що належить і розробляється Embarcadero Technologies. Embarcadero Delphi є частиною пакета Embarcadero RAD Studio і поставляється в п'яти редакціях: Starter, Professional, Enterprise, Ultimate і Architect

Embarcadero RAD Studio XE7 є повномасштабну систему розробки нативних додатків, орієнтованих не тільки на платформи iOS, Android, Mac і Windows, але і на переносні пристрої, які ґрунтуються на єдиній базі вихідних кодів. Embarcadero RAD Studio XE7 - це рішення, що дозволяє створювати продуктивні компілюють нативні додатки для будь-яких пристроїв, що володіють високим рівнем взаємодії з користувачем. Використання Embarcadero RAD Studio XE7 дає фахівцям можливість розширення своїх Windows-проектів до мобільних програм, які працюють з сервісами BaaS і даними масштабу організації.

За рахунок модернізації функцій VCL-бібліотеки, розробники отримують можливість удосконалювати, розвивати і створювати свої додатки для Windows. Embarcadero RAD Studio XE7 - це єдине в своїй галузі рішення, яке за допомогою підтримки Android, надає інтегровану середу розробки в режимі перегляду під управлінням Mac, iOS, Android і Windows. Користувачі мають можливість випускати нативні додатки для будь-яких платформ, в тому числі і Google Glass, на базі Object Pascal (Delphi) або

єдиного коду на C ++. Проста перекомпіляція дозволяє без зміни у вихідному коді перенести це ж додаток на інші платформи.

Переваги Embarcadero RAD Studio XE7:

- Bonus Pack. Додаткові продукти: набір стилів для FireMonkey Premium Style Packs і VCL, FastCube VCL 2 - компоненти і інструменти для аналізу даних;

- хмарні сервіси. За допомогою REST користувач отримує можливість підключення як до BAAS-провайдерам так і до хмарних сервісів для аутентифікації, отримання сервісів повідомлень, зберігання та інших;

- корпоративні мобільні сервіси (EMS). RAD Studio дозволяє задіяти готове вбудоване рішення, яке засноване на REST - Mobile Enterprise Application Platform (MEAP). Воно служить основою для розподілених і взаємодіючих додатків, забезпечує публікацію та хостинг спеціалізованих API, аналітику роботи клієнтів на різних пристроях, аутентифікацію і керування користувачами, доступ до даних і СУБД;

- Bluetooth App Tethering. Розробники отримують можливість розширювати функціонал своїх Windows-додатків шляхом їх зв'язування з мобільними додатками-компаньйонами за допомогою Bluetooth і WiFi;

- бібліотека паралельних обчислень. У версії Embarcadero RAD Studio XE7 передбачена вбудована бібліотека, призначена для підвищення продуктивності на багатоядерних системах VCL і FireMonkey-додатків;

- FireUI - призначений для різних пристроїв дизайнер користувацького інтерфейсу. Все, що необхідно - один раз створити користувацький інтерфейс відразу для всіх різних форм-факторів і ОС. Він врахує можливості, особливості та правила поведінки, які є типовими для кожної конкретної платформи;

- формування взаємодіючих додатків. Програма дає можливість з легкістю підключатися до хмарних сервісів і корпоративних баз даних, взаємодіяти з гаджетами, сенсорами і мобільними пристроями;

- розширення вже існуючих Windows-додатків. Тепер користувачі мають можливість забезпечувати взаємодію з дуже широким спектром нових

пристроїв - ношеними пристроями, сенсорами, гаджетами, планшетами, смартфонами - з єдиного середовища розробки.

### 3.2 Опис програми

Лістинг програми показаний в додатку А

Програма включає наступні функції: відкриття (створення в разі його відсутності), збереження файлу, додавання, зміна полів записи, видалення запису, перегляд вмісту файлу, обробка запитів. Вибір дії здійснюється за допомогою меню. Обробка помилок призначеного для користувача введення.

Для відкриття / збереження файлів використовувати стандартні діалогові вікна

Також на формі розміщені кнопки «Змінити», «Видалити», «Додати»

У базі аптека.dat зберігаються записи, що містять інформацію про ліки:

1) номер аптеки, 2) назву ліків, 3) ціна, 4) кількість.

Обробка наступних запитів:

- відомості про наявність ліків в певній аптеці;
- общуюю вартість всіх ліків у всіх аптеках;
- за назвою ліки видає список аптек, в яких воно є;
- знайти найдешевші ліки.

При відкритті програми відкривається головне вікно, де і відбувається введення даних.

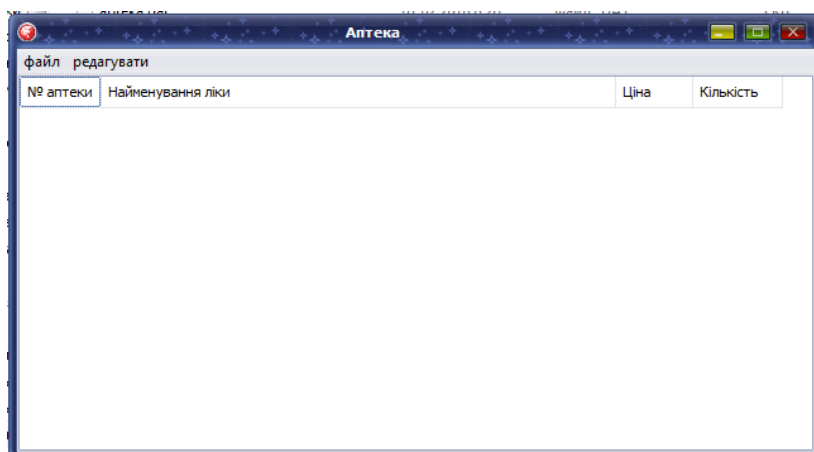


Рисунок 3.2.1 – Вікно програми

При наступному натисканні на владкові «Файл» з'являється допоміжне меню дозволяють відкрити, зберегти і інші дії з файлом

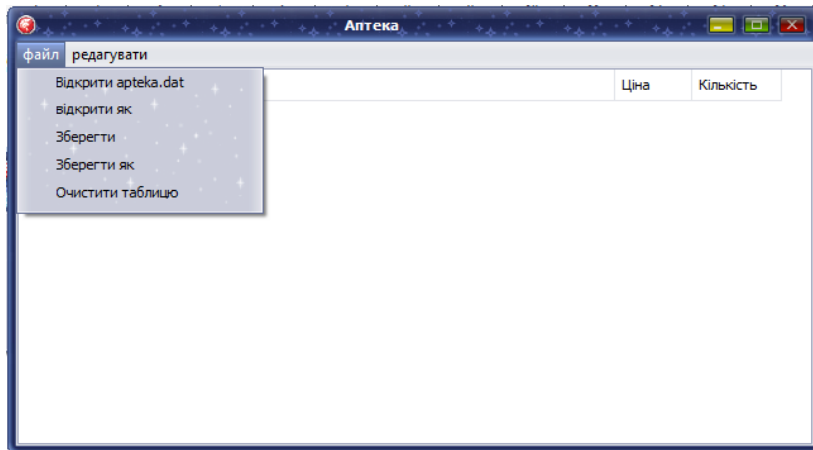
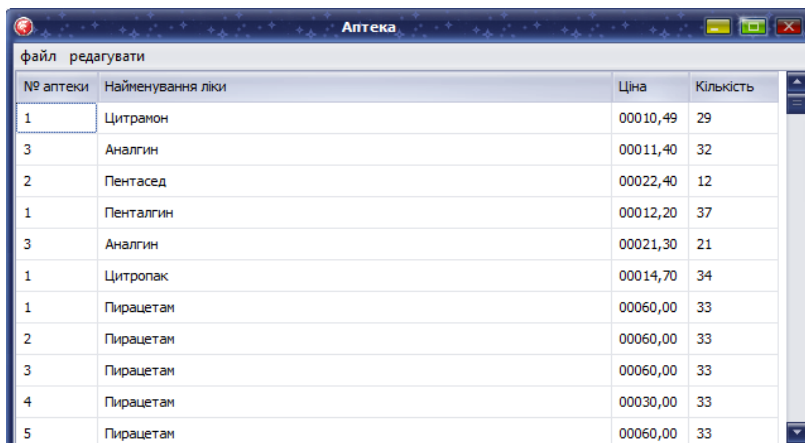


Рисунок 3.2.2 – Вікно програми

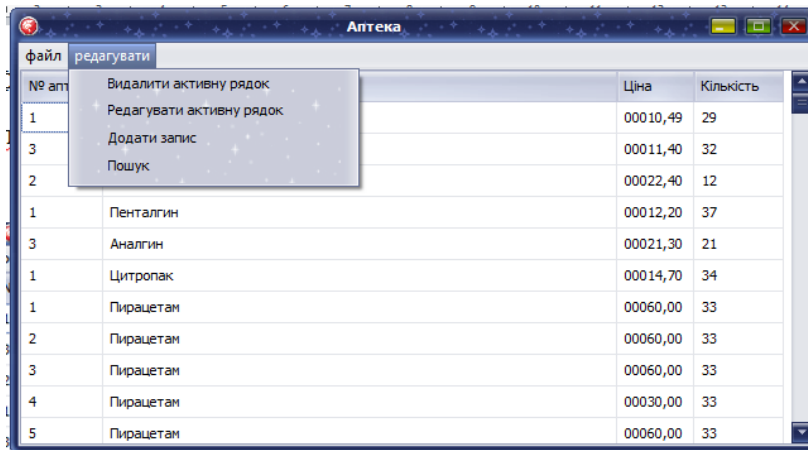
Вікно виведення результату дій. Після підрахунку вікно відкривається автоматично. У цьому вікні можна проводити збереження результату у форматі (.txt)



№ аптеки	Найменування ліки	Ціна	Кількість
1	Цитрамон	00010,49	29
3	Аналгин	00011,40	32
2	Пентасед	00022,40	12
1	Пенталгин	00012,20	37
3	Аналгин	00021,30	21
1	Цитропак	00014,70	34
1	Пирацетам	00060,00	33
2	Пирацетам	00060,00	33
3	Пирацетам	00060,00	33
4	Пирацетам	00030,00	33
5	Пирацетам	00060,00	33

Рисунок 3.2.3 – Вікно програми

Файл містить інформацію можна змінювати за допомогою вкладки «редагувати»



The screenshot shows the 'Аптека' application window with a table of pharmacy records. A context menu is open over the first row, showing options: 'Видалити активну рядок', 'Редагувати активну рядок', 'Додати запис', and 'Пошук'. The table has columns for '№ апт', 'Найменування ліки', 'Ціна', and 'Кількість'.

№ апт	Найменування ліки	Ціна	Кількість
1		00010,49	29
3		00011,40	32
2		00022,40	12
1	Пенталгин	00012,20	37
3	Аналгин	00021,30	21
1	Цитропак	00014,70	34
1	Пирацетам	00060,00	33
2	Пирацетам	00060,00	33
3	Пирацетам	00060,00	33
4	Пирацетам	00030,00	33
5	Пирацетам	00060,00	33

Рисунок 3.2.4 – Вікно довідка, короткий опис дії програми

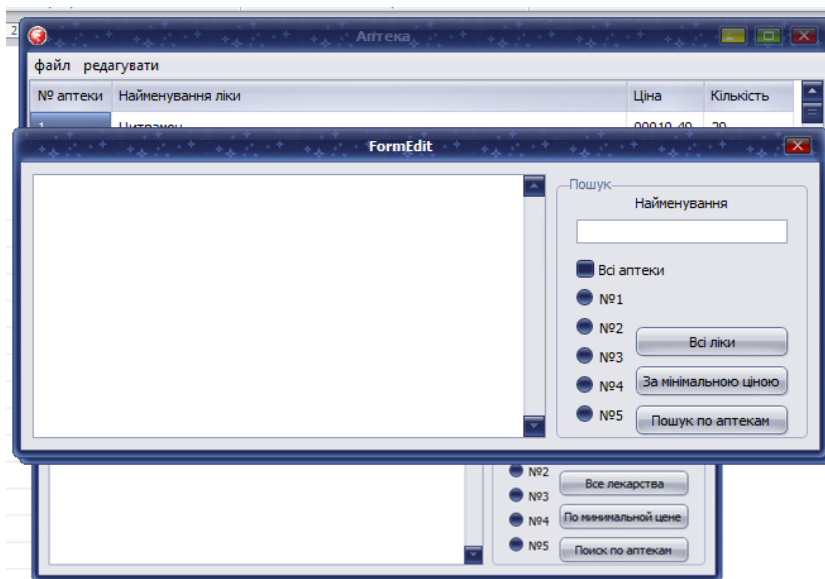
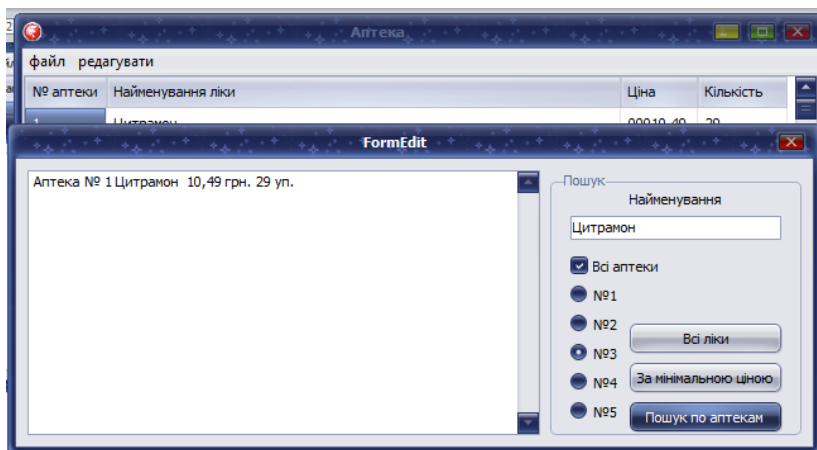


Рисунок 3.2.5 – Вікно програми



## Рисунок 3.2.6 – Вікно програми

### 3.3 Обробка помилок в програмі

При складанні програми важливо передбачити, щоб програма аналізувала можливі помилки, що виникають при її виконанні з вини користувача, інформувала його про це, підказуючи, що конкретно він зробив неправильно. При цьому можливо два підходи:

Запобігання помилок: програмно аналізувати вводяться або обчислюються дані і в разі, якщо вони можуть призводити до помилки, забезпечити, щоб програма інформувала користувача про необхідність коректного завдання даних.

Обробка помилок: в разі появи помилок, перехопити її, обробити і програмно відгукнутися на виниклу помилку.

При створенні програми треба поєднувати обидва підходи, застосовуючи в кожному конкретному випадку і для кожної можливої помилки той, який здається розробнику найбільш ефективним.

#### налагодження програм

При написанні програм користувач незалежно від його досвіду допускає ті чи інші помилки. Помилки - це об'єктивна неминучість або реальне втілення цієї неминучості. Умовно помилки можна поділити на три типи: помилки компіляції, виконання та логічні помилки.

#### помилки компіляції

Помилки компіляції виникають, якщо Borland Delphine може інтерпретувати введений код. Наприклад, при некоректному вводиті числа дужок, неправильному імені, неповному введенні інструкцій і т.д. Деякі з цих помилок виявляються Borland Delphi при завершенні набору рядки з інструкцією в редакторі коду і після натискання клавіші (Enter). Рядок, в якій міститься помилка, виділяється червоним кольором, і на екрані відображається діалогове вікно з повідомленням про можливу причину, що викликала помилку. Інші помилки компіляції виявляються перед виконанням



програми. Відмітимо, що Borland Delphi кожен раз автоматично компілює програму при її запуску на виконання.

#### помилки виконання

Помилки виконання виникають після успішної компіляції програми при її виконанні. Причинами таких помилок можуть бути, наприклад:

Некоректна інформація при зчитуванні файлу з диска.

Некоректні дані, введені користувачем, наприклад, потрібно число, а користувач вводить строкову інформацію.

Некоректне обчислення, наприклад розподіл на нуль.

У цих та подібних випадках на екрані відображається діалогове вікно Borland Delphi з повідомленням про номер помилки і можливу причину, її викликала.

#### Логічні помилки

Логічні помилки найважче виявити і усунути. Ці помилки не приводять до переривання виконання програми, тобто візуально все йде гладко і виглядає так, як ніби програма працює бездоганно. Але це тільки здається ідилія, тому що програма видає невірні результати. Локалізація логічних помилок пов'язана з ретельним аналізом алгоритму програми із залученням коштів налагодження Borland Delphi.

1 У ході виконання дипломної роботи було зроблено розроблена автоматизована інформаційна система обліку товару мережі аптек

2 У Delphi XE7 на базі середовища Embarcadero RAD Studio XE7 розроблена програма, яка дозволяє поліпшити роботу мережі аптек з обліку і реалізації товару .

Комп'ютерна програма виконує наступні основні функції:

- відомості про наявність ліків в певній аптеці;
- загальну вартість всіх ліків у всіх аптеках;
- за назвою ліки видає список аптек, в яких воно є;
- знаходить за запитом найдешевші ліки.

3 Одержані результати може використовуватися в роботі мережі аптек як безкоштовна альтернатива платним аналогам.

## СПИСОК ЛИТЕРАТУРА

1. Бастриков М.В. О.П.Пономарев Информационные технологии управления: Учебное пособие Институт «КВШУ».– Калининград: Изд-во Ин-та «КВШУ», 2005.– 140 с.
2. Болтенков В.И., Литвинов А.Л., Лычева Н.В. Конфигурирование и настройка автоматизированных информационных систем Учебное пособие. Белгород: Изд-во БелГУ, 2004.
3. Вендров, А.М. Проектирование программного обеспечения экономических информационных систем/ А.М. Вендров. - М.: Финансы и статистика, 2005. - 180с.
4. Гагарина, Л.Г. Разработка и эксплуатация автоматизированных информационных систем: учеб. ИД Форум: Инфра – М, 2007. – 384 с.
5. Емельянова Е.З., И. И. Попов, Т. Л. Партыка. – М. : ФОРУМ ИНФРА-М, 2012.Рудаков А. В. Технология разработки программных продуктов. Практикум : учеб. пособие для студ. учреждений сред. проф. образования / Устройство и функционирование информационных систем : учеб. пособие /
6. Мезенцев, К. Н. Автоматизированные информационные системы : учебник для студ. учреждений сред. проф. образования / К. Н. Мезенцев. - М.: Издательский центр «Академия», 2010.
7. Рудаков А. В. Технология разработки программных продуктов. Практикум : учеб. пособие для студ. учреждений сред. проф. образования / А. В. Рудаков, Г. Н. Федорова. - 3-е изд., испр. — М.: Издательский центр «Академия»; 2012.
8. Светлов М.Н, Светлова Г.Н Информационные технологии управления проектами: Учебное пособие. М.: ФГОУ ВПО РГАУ–МСХА им. К.А. Тимирязева, 2007. — 144 с.
9. Федорова Г. Н. Информационные системы : учебник для студ. учреждений сред. проф. образования. – 2-е изд., испр. – М. : ИЦ Академия, 2011
10. Фуфаев, Д. Э. Разработка и эксплуатация автоматизированных информационных систем : учебник для студ. сред. проф. образования / Д. Э. Фуфаев, Э. В. Фуфаев. - М. : Издательский центр «Академия», 2010

## Вхідні і отримані дані

```
unit Umain;

interface

uses
  Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes,
  Vcl.Graphics,
  Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.Menus, Vcl.Grids, Vcl.StdCtrls;

type
  TFormMain = class(TForm)
    MainMenu1: TMainMenu;
    N1: TMenuItem;
    N3: TMenuItem;
    OpenFileDialog1: TOpenDialog;
    SaveDialog1: TSaveDialog;
    StringGrid1: TStringGrid;
    N5: TMenuItem;
    N6: TMenuItem;
    N4: TMenuItem;
    N8: TMenuItem;
    N9: TMenuItem;
    N7: TMenuItem;
    N10: TMenuItem;
    N11: TMenuItem;
    N2: TMenuItem;
    procedure N2Click(Sender: TObject);
    procedure N3Click(Sender: TObject);
    procedure N4Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure N5Click(Sender: TObject);
    procedure N6Click(Sender: TObject);
    procedure N8Click(Sender: TObject);
    procedure N9Click(Sender: TObject);
    procedure StringGrid1Click(Sender: TObject);
    procedure N7Click(Sender: TObject);
    procedure N11Click(Sender: TObject);
  private
    { Private declarations }
  public
    type
      TPupil = record
        Nomer: string[20]; // указываем фиксированную длину строки
        Surname: string[20]; //
        Prise: string[20];
        Kolvo: string[20];
      end;
    { Public declarations }
  end;

var
  FormMain: TFormMain;
```

implementation

uses unedit;

{ \$R \*.dfm }

procedure TFormMain.FormCreate(Sender: TObject);

begin

{Задание числа строк и столбцов в таблицах}

StringGrid1.ColWidths[0]:=64;

StringGrid1.ColWidths[1]:=406;

StringGrid1.ColWidths[2]:=60;

StringGrid1.ColWidths[3]:=70;

StringGrid1.RowCount:=1;

StringGrid1.ColCount:=4;

StringGrid1.Cells[0,0]:='№ аптеки';

StringGrid1.Cells[1,0]:='Найменування ліки';

StringGrid1.Cells[2,0]:='Ціна';

StringGrid1.Cells[3,0]:='Кількість';

end;

procedure TFormMain.N1Click(Sender: TObject);

begin

formedit.Memo1.Visible:=false;

if StringGrid1.RowCount<=1 then begin

showmessage('Файл не завантажено!');

exit;

end;

formedit.show;

FormEdit.EditNaptEDit.Text:=FormMain.StringGrid1.Cells[0,FormMain.StringGrid1.Row];

FormEdit.EditNpr.Text:=FormMain.StringGrid1.Cells[1,FormMain.StringGrid1.Row];

FormEdit.MaskEditPrEdit.Text:=FormMain.StringGrid1.Cells[2,FormMain.StringGrid1.Row];

FormEdit.EditKolvoEdit.Text:=FormMain.StringGrid1.Cells[3,FormMain.StringGrid1.Row];

end;

procedure TFormMain.N2Click(Sender: TObject);

begin

if StringGrid1.RowCount<=1 then begin

showmessage('Файл не завантажено!');

exit;

end;

formedit.show;

formedit.memo1.Visible:=true;

end;

procedure TFormMain.N3Click(Sender: TObject);

var

buttonSelected : Integer;

ib: byte;

F: file of TPupil; // объявляем файловую переменную

P: array of TPupil;

i: integer;

begin

if StringGrid1.RowCount<=1 then begin

showmessage('Таблица порожня!');

exit;

end;

```

buttonSelected := MessageDlg('Видалити активну рядок містить
'+StringGrid1.Cells[StringGrid1.Col,StringGrid1.Row]+'
'+sLineBreak+sLineBreak+'Yes'+sLineBreak+sLineBreak+'No',mtCustom,
    [mbYes,mbNo], 0);
if buttonSelected = mrYes then begin
    //удаление строки
    if StringGrid1.RowCount = 2 then StringGrid1.Rows[1].Clear;
    if StringGrid1.RowCount > 2 then
        begin
            for ib := StringGrid1.Row to StringGrid1.RowCount-1 do
                StringGrid1.Rows[ib] := StringGrid1.Rows[ib+1];
            StringGrid1.RowCount := StringGrid1.RowCount-1;
        end;
    //запись в промежуточный файл аптекаTEMP.dat
    AssignFile(F,'аптекаTEMP.dat');// связываем переменную с файлом на диске
    Rewrite(f); // открываем файл, перезаписываем его содержимое
    i:=1; //запускаем счетчик записей
    while i<StringGrid1.RowCount do //пройдем по всем строкам в таблице
        begin
            SetLength(P, i+1); // привязал длину массива к I
            //считываем данные из ячеек таблицы в переменную типа TPupil
            P[i].Nomer:=StringGrid1.Cells[0,i];
            P[i].Surname:=StringGrid1.Cells[1,i];
            P[i].Prise:=StringGrid1.Cells[2,i];
            P[i].Kolvo:=StringGrid1.Cells[3,i];
            write(f,P[i]); //записываем данные в файл
            inc(i); //увеличиваем счетчик
        end;
    CloseFile(f); //обязательно закрываем файл!
    CopyFile('аптекаTEMP.dat','аптека.dat',false);//по условию задачи
end;
end;

```

```

procedure TFormMain.N4Click(Sender: TObject);
var
    F: file of TPupil;// объявляем файловую переменную
    P: array of TPupil;
    i: integer;
begin
    try
        if SaveDialog1.Execute then begin
            AssignFile(F,SaveDialog1.FileName);// связываем переменную с файлом на диске
            Rewrite(f); // открываем файл, перезаписываем его содержимое
            i:=1; //запускаем счетчик записей
            while i<StringGrid1.RowCount do //пройдем по всем строкам в таблице
                begin
                    SetLength(P, i+1); // привязал длину массива к I
                    //считываем данные из ячеек таблицы в переменную типа TPupil
                    P[i].Nomer:=StringGrid1.Cells[0,i];
                    P[i].Surname:=StringGrid1.Cells[1,i];
                    P[i].Prise:=StringGrid1.Cells[2,i];
                    P[i].Kolvo:=StringGrid1.Cells[3,i];
                    write(f,P[i]); //записываем данные в файл
                    inc(i); //увеличиваем счетчик
                end;
        end;
    end;
end;

```

```

    CloseFile(f); //обязательно закрываем файл!
    end else exit;
except on E: Exception do
    raise Exception.Create(E.Message+Slinebreak+slinebreak+'Можливо відсутність або
пошкодження файл аптека.dat');
end;
end;

procedure TFormMain.N5Click(Sender: TObject);
begin
    formedit.Show;
    formedit.Memo1.Visible:=false;
end;

procedure TFormMain.N6Click(Sender: TObject);
var
    F: file of TPupil; // объявляем файловую переменную
    P: array of TPupil;
    i: integer;
begin
    {Задание числа строк и столбцов в таблицах}
    StringGrid1.ColWidths[0]:=64;
    StringGrid1.ColWidths[1]:=406;
    StringGrid1.ColWidths[2]:=60;
    StringGrid1.ColWidths[3]:=70;

    StringGrid1.RowCount:=1;
    StringGrid1.ColCount:=4;
    StringGrid1.Cells[0,0]:='№ аптеки';
    StringGrid1.Cells[1,0]:='Найменування ліки';
    StringGrid1.Cells[2,0]:='Ціна';
    StringGrid1.Cells[3,0]:='Кількість';

    try
    if FileExists('apтека.dat') then //проверяем, существует ли файл
    begin
        i:=1;
        AssignFile(F,'apтека.dat');//связываем переменную с файлом на диске
        Reset(f); //открываем файл
        //будем читать содержимое, пока указатель на достигнет конца файла
        while not(EOF(F)) do
        begin
            SetLength(P, i+1); // привязал длину массива к I
            read(f,P[i]);//считываем данные в переменную типа TPupil
            // указатель смещается на следующую запись в файле
            StringGrid1.RowCount:=StringGrid1.RowCount+1; //увеличиваем количество строк в
таблице
            // заносим данные из полей записи в ячейки таблицы в новую строку
            StringGrid1.Cells[0,StringGrid1.RowCount-1]:=P[i].Nomer;
            StringGrid1.Cells[1,StringGrid1.RowCount-1]:=P[i].Surname;
            StringGrid1.Cells[2,StringGrid1.RowCount-1]:=P[i].Prise;
            StringGrid1.Cells[3,StringGrid1.RowCount-1]:=P[i].Kolvo;
            inc(i);//увеличиваем счетчик записей
        end;
        CloseFile(f); //обязательно закрываем файл!
    end;
end;

```

```

StringGrid1.FixedRows:=1;
except on E: Exception do
  raise Exception.Create(E.Message+Slinebreak+slinebreak+'Можливо відсутність або
пошкодження файл аптека.dat');
end;
end;

procedure TFormMain.N7Click(Sender: TObject);
var
  F: file of TPupil; // об'являем файловую переменную
  P: array of TPupil;
  i: integer;
begin
  if StringGrid1.RowCount<=1 then begin
    showmessage('Таблиця порожня!');
    exit;
  end;
  try
    AssignFile(F,'аптека.dat');// связываем переменную с файлом на диске
    Rewrite(f); // открываем файл, перезаписываем его содержимое
    i:=1; //запускаем счетчик записей
    while i<StringGrid1.RowCount do //пройдем по всем строкам в таблице
      begin
        SetLength(P, i+1); // привязал длину массива к I
        //считываем данные из ячеек таблицы в переменную типа TPupil
        P[i].Nomer:=StringGrid1.Cells[0,i];
        P[i].Surname:=StringGrid1.Cells[1,i];
        P[i].Prise:=StringGrid1.Cells[2,i];
        P[i].Kolvo:=StringGrid1.Cells[3,i];
        write(f,P[i]); //записываем данные в файл
        inc(i); //увеличиваем счетчик
      end;
    CloseFile(f); //обязательно закрываем файл!
  except on E: Exception do
    raise Exception.Create(E.Message+Slinebreak+slinebreak+'Можливо недостатньо ОЗУ');
  end;
end;

procedure TFormMain.N8Click(Sender: TObject);
var
  F: file of TPupil; // об'являем файловую переменную
  P: array of TPupil;
  i: integer;
begin
  {Задание числа строк и столбцов в таблицах}
  StringGrid1.ColWidths[0]:=64;
  StringGrid1.ColWidths[1]:=406;
  StringGrid1.ColWidths[2]:=60;
  StringGrid1.ColWidths[3]:=70;

  StringGrid1.RowCount:=1;
  StringGrid1.ColCount:=4;
  StringGrid1.Cells[0,0]:='№ аптеки';
  StringGrid1.Cells[1,0]:='Найменування ліки';
  StringGrid1.Cells[2,0]:='Ціна';
  StringGrid1.Cells[3,0]:='Кількість';

```



```

try
  If OpenFileDialog1.Execute then begin
    if FileExists('apteka.dat') then //проверяем, существует ли файл
    begin
      i:=1;
      AssignFile(F,OpenDialog1.FileName);//связываем переменную с файлом на диске
      Reset(f); //открываем файл
      //будем читать содержимое, пока указатель на достигнет конца файла
      while not(EOF(F)) do
      begin
        SetLength(P, i+1); // привязал длину массива к I
        read(f,P[i]);//считываем данные в переменную типа TPupil
        // указатель смещается на следующую запись в файле
        StringGrid1.RowCount:=StringGrid1.RowCount+1; //увеличиваем количество строк в
таблице
        // заносим данные из полей записи в ячейки таблицы в новую строку
        StringGrid1.Cells[0,StringGrid1.RowCount-1]:=P[i].Nomer;
        StringGrid1.Cells[1,StringGrid1.RowCount-1]:=P[i].Surname;
        StringGrid1.Cells[2,StringGrid1.RowCount-1]:=P[i].Prise;
        StringGrid1.Cells[3,StringGrid1.RowCount-1]:=P[i].Kolvo;
        inc(i);//увеличиваем счетчик записей
      end;
      CloseFile(f); //обязательно закрываем файл!
    end;
  end;
  StringGrid1.FixedRows:=1;
  except on E: Exception do
    raise Exception.Create(E.Message+Slinebreak+slinebreak+'Можливо відсутність або
пошкодження файл аптека.dat');
  end;
end;

procedure TFormMain.N9Click(Sender: TObject);
begin
  {Задание числа строк и столбцов в таблицах}
  StringGrid1.ColWidths[0]:=64;
  StringGrid1.ColWidths[1]:=406;
  StringGrid1.ColWidths[2]:=60;
  StringGrid1.ColWidths[3]:=70;

  StringGrid1.RowCount:=1;
  StringGrid1.ColCount:=4;
  StringGrid1.Cells[0,0]:='№ аптеки';
  StringGrid1.Cells[1,0]:='Найменування ліки';
  StringGrid1.Cells[2,0]:='Ціна';
  StringGrid1.Cells[3,0]:='Кількість';
end;

procedure TFormMain.StringGrid1Click(Sender: TObject);
begin
  if StringGrid1.RowCount<=1 then exit;
  FormEdit.EditNaptEDit.Text:=FormMain.StringGrid1.Cells[0,FormMain.StringGrid1.Row];
  FormEdit.EditNpr.Text:=FormMain.StringGrid1.Cells[1,FormMain.StringGrid1.Row];
  FormEdit.MaskEditPrEdit.Text:=FormMain.StringGrid1.Cells[2,FormMain.StringGrid1.Row];
  FormEdit.EditKolvoEdit.Text:=FormMain.StringGrid1.Cells[3,FormMain.StringGrid1.Row];
end;

```

end.

unit UnitEdit;

interface

uses

Winapi.Windows, Winapi.Messages, System.SysUtils, System.Variants, System.Classes,  
Vcl.Graphics,  
Vcl.Controls, Vcl.Forms, Vcl.Dialogs, Vcl.StdCtrls, Vcl.Mask;

type

```
TFormEdit = class(TForm)
  GroupBox1: TGroupBox;
  Button1: TButton;
  EditKolvo: TEdit;
  Label4: TLabel;
  MaskEditPrice: TMaskEdit;
  Label3: TLabel;
  EditNamePR: TEdit;
  EditNAPT: TEdit;
  Label2: TLabel;
  Label1: TLabel;
  GroupBox2: TGroupBox;
  EditNPr: TEdit;
  Label5: TLabel;
  Label6: TLabel;
  EditKolvoEdit: TEdit;
  Label7: TLabel;
  MaskEditPrEdit: TMaskEdit;
  Label8: TLabel;
  EditNaptEDit: TEdit;
  Button4: TButton;
  GroupBox3: TGroupBox;
  RadioButton4: TRadioButton;
  RadioButton5: TRadioButton;
  RadioButton3: TRadioButton;
  RadioButton2: TRadioButton;
  RadioButton1: TRadioButton;
  CheckBox1: TCheckBox;
  Edit1: TEdit;
  Button2: TButton;
  Label9: TLabel;
  Memo1: TMemo;
  Button3: TButton;
  Button5: TButton;
  procedure Button1Click(Sender: TObject);
  procedure EditKolvoKeyPress(Sender: TObject; var Key: Char);
  procedure Button2Click(Sender: TObject);
  procedure MaskEditPriceChange(Sender: TObject);
  procedure MaskEditPrEditChange(Sender: TObject);
  procedure Button4Click(Sender: TObject);
  procedure FormClose(Sender: TObject; var Action: TCloseAction);
  procedure EditNAPTKeyPress(Sender: TObject; var Key: Char);
  procedure Button3Click(Sender: TObject);
```

```

    procedure Button5Click(Sender: TObject);
private
    { Private declarations }
public
    type
        TPupil = record
            Nomer: string[20]; // указываем фиксированную длину строки
            Surname: string[20]; //
            Prise: string[20];
            Kolvo: string[20];
        end;
    { Public declarations }
end;

var
    FormEdit: TFormEdit;

implementation

{$R *.dfm}
Uses umain;
procedure TFormEdit.Button1Click(Sender: TObject);
var
    F: file of TPupil; // объявляем файловую переменную
    P: TPupil;
    i: integer;
begin
    if ((EditNAPT.Text="") or (EditNamePR.Text="") or (MaskEditPrice.Text=',') or
        (EditKolvo.Text="")) then begin
        Showmessage('Введите все 4 параметра, №аптеки, наименования, цена, количество.');
```

exit;

```

    end;

AssignFile(F,'apteka.dat'); // связываем переменную с файлом на диске
if FileExists('apteka.dat') then //проверяем, существует ли файл
begin
    Reset(f); //если файл существует, открываем его
    Seek(f,system.FileSize(f)); //устанавливаем указатель на конец файла
    //после последней записи
end else Rewrite(f); //если файла не существует, создаем его
    //считываем данные из Edit-ов в поля записи
    P.Nomer:=EditNAPT.Text;
    P.Surname:=EditNamePR.Text;
    P.Prise:=MaskEditPrice.Text;
    P.Kolvo:=EditKolvo.Text;
    write(F,P); //записываем значение переменной P файл F на текущую позицию
    //при этом позиция указателя сместится на одну вправо, то есть в конец файла
    FormMain.StringGrid1.RowCount:=FormMain.StringGrid1.RowCount+1; //добавим одну
    пустую строку в таблицу
    //заносим данные из переменной типа TPupil в ячейки таблицы
    FormMain.StringGrid1.Cells[0,FormMain.StringGrid1.RowCount-1]:=P.Nomer;
    FormMain.StringGrid1.Cells[1,FormMain.StringGrid1.RowCount-1]:=P.Surname;
    FormMain.StringGrid1.Cells[2,FormMain.StringGrid1.RowCount-1]:=P.Prise;
    FormMain.StringGrid1.Cells[3,FormMain.StringGrid1.RowCount-1]:=P.Kolvo;
    closefile(f); //обязательно закрываем файл по окончании работы с ним!
end;

```

```

procedure TFormEdit.Button2Click(Sender: TObject);
var i, j: Integer;
begin
//поиск по полю наименование
formedit.memo1.Visible:=true;
memo1.Clear;
if ((Edit1.Text='') or ((checkbox1.Checked=false) and (Radiobutton1.Checked=false) and
(Radiobutton2.Checked=false) and (Radiobutton3.Checked=false) and
(Radiobutton4.Checked=false) and (Radiobutton5.Checked=false))) then begin
Showmessage('Не вказаний діапазон пошуку. Введіть найменування препарату або вибрати
інший тип аптеки');
exit;
end;
if checkbox1.Checked=true then begin
with FormMain.StringGrid1 do
for i:=1 to RowCount-1 do //Заголовки строк не трогаем
for j:=1 to 1 do begin
if Cells[j, i]=Edit1.Text then memo1.Lines.Add('Аптека № '+Cells[0, i]+' '+Cells[1, i]+'
'+FloatToStr(StrToFloat(Cells[2, i]))+' грн. '+Cells[3, i]+' уп. ');
end;
end;
if Radiobutton1.Checked=true then begin
with FormMain.StringGrid1 do
for i:=1 to RowCount-1 do //Заголовки строк не трогаем
for j:=1 to 1 do begin
if ((Cells[j, i]=Edit1.Text) and (Cells[0, i]='1')) then memo1.Lines.Add('Аптека № '+Cells[0,
i]+' '+Cells[1, i]+' '+FloatToStr(StrToFloat(Cells[2, i]))+' грн. '+Cells[3, i]+' уп. ');
end;
end;
if Radiobutton2.Checked=true then begin
with FormMain.StringGrid1 do
for i:=1 to RowCount-1 do //Заголовки строк не трогаем
for j:=1 to 1 do begin
if ((Cells[j, i]=Edit1.Text) and (Cells[0, i]='2')) then memo1.Lines.Add('Аптека № '+Cells[0,
i]+' '+Cells[1, i]+' '+FloatToStr(StrToFloat(Cells[2, i]))+' грн. '+Cells[3, i]+' уп. ');
end;
end;
if Radiobutton3.Checked=true then begin
with FormMain.StringGrid1 do
for i:=1 to RowCount-1 do //Заголовки строк не трогаем
for j:=1 to 1 do begin
if ((Cells[j, i]=Edit1.Text) and (Cells[0, i]='3')) then memo1.Lines.Add('Аптека № '+Cells[0,
i]+' '+Cells[1, i]+' '+FloatToStr(StrToFloat(Cells[2, i]))+' грн. '+Cells[3, i]+' уп. ');
end;
end;
if Radiobutton4.Checked=true then begin
with FormMain.StringGrid1 do
for i:=1 to RowCount-1 do //Заголовки строк не трогаем
for j:=1 to 1 do begin
if ((Cells[j, i]=Edit1.Text) and (Cells[0, i]='4')) then memo1.Lines.Add('Аптека № '+Cells[0,
i]+' '+Cells[1, i]+' '+FloatToStr(StrToFloat(Cells[2, i]))+' грн. '+Cells[3, i]+' уп. ');
end;
end;
if Radiobutton5.Checked=true then begin

```

```

with FormMain.StringGrid1 do
  for i:=1 to RowCount-1 do //Заголовки строк не трогаем
    for j:=1 to 1 do begin
      if ((Cells[j, i]=Edit1.Text) and (Cells[0, i]='5')) then memo1.Lines.Add('Аптека № '+Cells[0,
i]+' '+Cells[1, i]+' '+FloatToStr(StrToFloat(Cells[2, i]))+' грн. '+Cells[3, i]+' уп. ');
      end;
    end;

end;

procedure TFormEdit.Button3Click(Sender: TObject);
var
  i, j: Integer;
  x: real;
  list: TStringList;
  str: String;
begin
  memo1.Visible:=True;;
  x:=1000000000;
  list := TStringList.Create;
  list.Sorted := True;
  list.Duplicates := dupIgnore;
  with FormMain.StringGrid1 do
    for i:=1 to RowCount-1 do //Заголовки строк не трогаем
      for j:=1 to 1 do begin
        if Cells[j, i]=Edit1.Text then begin
          if StrToFloat(Cells[2, i])<x then begin x:= StrToFloat(Cells[2, i]);
            memo1.Clear;
            str := Trim('Аптека № '+Cells[0, i]+' '+Cells[1, i]+' '+FloatToStr(StrToFloat(Cells[2, i]))+'
грн. '+Cells[3, i]+' уп. ');
            list.Add(str);
          end;
          if StrToFloat(Cells[2, i])=x then begin
            str := Trim('Аптека № '+Cells[0, i]+' '+Cells[1, i]+' '+FloatToStr(StrToFloat(Cells[2, i]))+'
грн. '+Cells[3, i]+' уп. ');
            list.Add(str);
          end;
        end;
      end;
    end;

  Memo1.Lines.Assign(list);
  list.Free;
end;

procedure TFormEdit.Button4Click(Sender: TObject);
var
  F: file of TPupil;// объявляем файловую переменную
  P: array of TPupil;
  i: integer;
begin
  if ((EditNaptEDit.Text='') or (EditNpr.Text='') or (MaskEditPrEdit.Text=',' ) or
(EditKolvoEdit.Text='')) then begin
    Showmessage('Помилка введення, перевірте введені значення і повторіть');
  exit;
end;

```

```
FormMain.StringGrid1.Cells[0,FormMain.StringGrid1.Row]:= EditNaptEdit.Text;
FormMain.StringGrid1.Cells[1,FormMain.StringGrid1.Row]:= EditNpr.Text;
FormMain.StringGrid1.Cells[2,FormMain.StringGrid1.Row]:= MaskEditPrEdit.Text;
FormMain.StringGrid1.Cells[3,FormMain.StringGrid1.Row]:= EditKolvoEdit.Text;
```

```
//запись изменений в файл
```

```
AssignFile(F,'apteka.dat');// связываем переменную с файлом на диске
Rewrite(f); // открываем файл, перезаписываем его содержимое
i:=1; //запускаем счетчик записей
while i<FormMain.StringGrid1.RowCount do //пройдем по всем строкам в таблице
begin
SetLength(P, i+1); // привязал длину массива к I
//считываем данные из ячеек таблицы в переменную типа TPupil
P[i].Nomer:=FormMain.StringGrid1.Cells[0,i];
P[i].Surname:=FormMain.StringGrid1.Cells[1,i];
P[i].Prise:=FormMain.StringGrid1.Cells[2,i];
P[i].Kolvo:=FormMain.StringGrid1.Cells[3,i];
write(f,P[i]); //записываем данные в файл
inc(i); //увеличиваем счетчик
end;
CloseFile(f); //обязательно закрываем файл!
end;
```

```
procedure TFormEdit.Button5Click(Sender: TObject);
var
i, j: Integer;
x: double;
begin
x:=0;
Memo1.Clear;
memo1.Visible:=True;;
with FormMain.StringGrid1 do
for i:=1 to RowCount-1 do //Заголовки строк не трогаем
for j:=2 to 2 do x:=x+StrToFloat(Cells[2, i]);
Memo1.Lines.Text:=(Загалом в усіх аптеках всіх ліків на суму: '+FloatToStr(x)+' грн. ');
end;
```

```
procedure TFormEdit.EditKolvoKeyPress(Sender: TObject; var Key: Char);
begin
if not ((Key<' ') or ((Key>='0')and(Key<='9')))
then
Key := chr(0);
end;
```

```
procedure TFormEdit.EditNAPTKeyPress(Sender: TObject; var Key: Char);
begin
if not ((Key<' ') or ((Key>='0')and(Key<='9')))
then
Key := chr(0);
end;
```

```
procedure TFormEdit.FormClose(Sender: TObject; var Action: TCloseAction);
begin
memo1.Visible:=false;
memo1.Clear;
end;
```

```
procedure TFormEdit.MaskEditPrEditChange(Sender: TObject);
  var A1: double;
begin
  if MaskEditPrEdit.Text='' then exit;
  try
    A1 := StrToFloat(MaskEditPrEdit.Text);
  except on EConvertError do
    ShowMessage('Ви ввели помилкове число; повторіть введення');
  end;
end.
```