Вступ 3

1. Аналіз технічного завдання 4

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

2

**КР СН-08-033 ПЗ**

Розроб.

Савчук В.О.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Зміст**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

1. Обґрунтування алгоритму і структури програми 5
2. Розробка програми 6

4. UML діаграма класів 10

5. Тестування програми та результат її виконання 12

Висновок 17

Список використаної літератури 18

**ВСТУП**

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

3

**КР СН-08-033 ПЗ**

Розроб.

Савчук В.О.

.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Вступ**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

Під час роботи людей які працюють і торговій сфері виникає проблема обліку наявного товару у складах, труднощі і зменшення швидкості роботи персоналу при перераховуванню наявного товару на складі, щоб дати відповідь про те що залишилося, його ціну та ін. На даний період комп’ютеризована велика кількість організацій, правда комп’ютери не завжди мають високу швидкості обробки даних. Тому виникає проблема у простій програмі яка не буде займати багато місця на комп’ютері та можливість ПК швидко обробити її. Така програма повинна бути найбільш простою у користуванні, що забезпечить використання програми найменш кваліфікованим користувачем. Також було б непогано виконати простий інтерфейс для цієї програми, і просту логіку внесення і отримання потрібних даних.

1. **АНАЛІЗ ТЕХНІЧНОГО ЗАВДАННЯ**

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

4

**КР СН-08-033 ПЗ**

Розроб.

Савчук В.О

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Аналіз технічного завдання**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

Завдання даної курсової роботи полягає у написанні програми для створення бази даних медикаментів ввіділу лікарні. Також за допомогою цієї програми можна легко і доступно для оператора додавати в базу даних ліки, дізнатись їх кількість, вносити зміни в кількість доступного препарату.

Для забезпечення виконання поставленого завдання, потрібно розробити функції, а в них описати їх призначення:

а) внесення в базу даних нових ліків;

б) виведення всіх ліків загалом;

в) пошук ліків по базі даних, за різними критеріями;

г) функція зміни і виведення кількості доступного препарату;

д) функція сортування ліків за критеріями.

1. **ОБГРУНТУВАННЯ АЛГОРИТМУ І СТРУКТУРИ ПРОГРАМИ**

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

5

**КР СН-08-033 ПЗ**

Розроб.

Савчук В.О

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Обгрунтування алгоритму**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

Для того щоб реалізувати виконання поставленого завданння у програмі використовується основний клас - class Pharmacy, що має відповідні методи, за допомогою яких обробляється подана інформація. Клас описує поля, методи, модифікатори, конструктор і деструктор.

private - опис прихованих полів і методів класу;

public- опис доступних полів і методів класу;

До класу private відносяться поля і методи: px, n, void remove ().

void remove () - допоміжна операція до функції видалення рядка в масиві.

Стуктуру Apteka зручніше використовувати для запису даних у файл і зчитування з нього.

1. **РОЗРОБКА ПРОГРАМИ**

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

6

**КH СН-08-033 ПЗ**

Розроб.

Савчук В.О.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Розробка програми**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

Спочатку в мене є головна частина програми і вона складається з оголошення бібліотек, структури та клас аптека, підключення підпрограм.

int main() - Ця функція є головною підпрограмою і складається з виведення головного меню і вибору операції.

Клас Pharmacy використовує методи:

void create() -Функція, яка відповідає за створення файлу, введення ліків у базу даних у цей файл. Ця функція виводить запит на заповнення нового рядка таблиці бази даних. Після введення даних (назва препарату, тип препарату, ціна і кількість) програма питає чи бажаємо ми додати ще нові препарат.

void Pharmacy::create()

{

int i;

px=new Apteka[n];

if(px==NULL)

{

cout<<"No Memory";

getch();

exit(1);

}

n=0;

adding();

do

{

cout<<"To add yet?\n";

cout<<"1-Yes\n";

cout<<"2-No\n";

cin>>i;

if(i==1)adding();

}while(i==1);

}

void reading()- Функція, які відповідає за вікриття користувацького файлу для читання.

void Pharmacy::reading()

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

7

**КH СН-08-033 ПЗ**

Розроб.

Савчук В.О.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Розробка програми**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

{

ifstream in;

char file[N];

int i; Apteka u;

cout<<"File name:\n";

cin>>file;

in.open(file);

if(in.fail())

{

cout<<file<<"Don't open";getch();exit(1);

}

n=0;

do

{

in>>u.Medication>>u.Type>>u.Amount>>u.Price;

n++;

}while(in.good());

void output()- Функція, які відповідає за виведення всієї бази даних із файла. Функція послідовно виводить всі доступні ліки на даний момент залежності від порядку введення його в базу.

Void find()- Функція, яка відповідає за пошук ліків. При виборі відповідного пункту меню нам виводяться критерії пошуку, а конкретно: за назвою, за типом та за ціною . Для кожного критерію є своя змінна, яку вибирає користувач для пошуку. Якщо змінна введена, то користувачу буде запропоновано ввести інформацію по якій буде проводитися порівняння даних, які є в аптечному складі з веденими користувачем. Якщо нічого не знайдено, то користувач побачить повідомлення про відсутність даних. Після введення всіх даних по яких користувач забажав провести пошук програма порівнює дані бази даних і ті дані, що ввів користувач. Якщо є збіги, то знайдені дані виводяться на екран.

Void record()- Функція, яка створює і записує інформацію у файл.

void Pharmacy::record()

{

int i;

ofstream out;

char file[N];

cout<<"Name file save:\n";

cin>>file;

cout.open(file);

if(out==NULL)

{

cout<<file<<"Don' create";

getch();

exit(1);

}

Void adding () – Функція додає до бази даних нові медикаменти.

void Pharmacy::adding()

{ Apteka \*s;int i;

s=new Apteka[n+1];

for(i=0;i<n;i++)

{

s[i]=px[i];

cout<<"Enter information:\n";

cout<<"Medication:\n";cin>>s[n].Medication;

cout<<"Type:\n";cin>>s[n].Type;

cout<<"Amount:\n";cin>>s[n].Amount;

cout<<"Price:\n";cin>>s[n].Price;

}

n++;

delete[]px;

px=s;

cout<<"Add!";

getch();

}

Void correct() – дозволяє змінювати кількість і ціну препарату. Після відповідних змін користувач отримує на екран базу даних із зміненими параметрами.

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

8

**КH СН-08-033 ПЗ**

Розроб.

Савчук В.О.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Розробка програми**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

Void remove() – вилучає вказаний користувачем медикамент із бази даних.

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

9

**КH СН-08-033 ПЗ**

Розроб.

Савчук В.О.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Розробка програми**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

void Pharmacy::Remove(int h)

{

int i;

if(h<0||h>=n)

{

cout<<"Does not exist";

getch();

return;

}

for(i=h+1;i<n;i++)

{

px[i-1]=px[i];

n--;

}

}

Void sorting() – Функція, яка відповідає за сортування ліків. При виборі відповідного пункту меню нам виводяться критерії сортування, а конкретно: за назвою, за типом, за кількість та за ціною. Після введення всіх даних по яких користувач забажав провести сортування, програма порівнює дані, що ввів користувач. Знайдені дані виводяться на екран.

1. **UML ДІАГРАМА КЛАСІВ**

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

10

**КР СН-08-033 ПЗ**

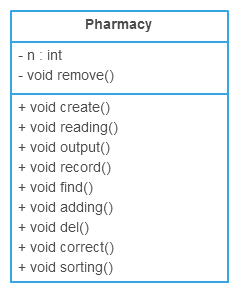


Рис.4.1 – Діаграма класів

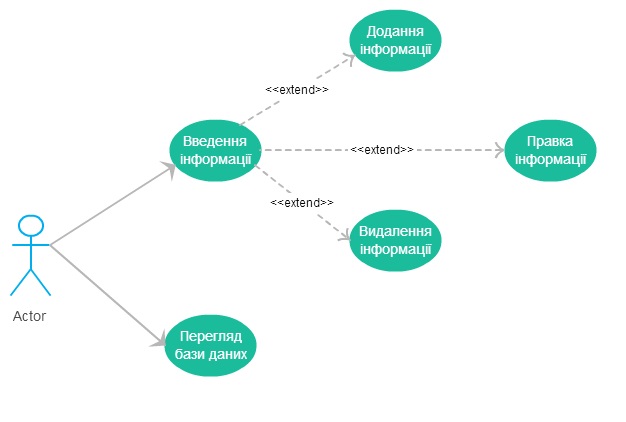


Рис. 4.2 – Діаграма прецедентів

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

11

**КР СН-08-033 ПЗ**

1. **ТЕСТУВАННЯ ПРОГРАМИ ТА РЕЗУЛЬТАТ ЇЇ ВИКОНАННЯ**

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

12

**КР СН-08-033 ПЗ**

При вході в програму з'являється меню з вибором пунктів меню, доступних для будь-якого користувача:

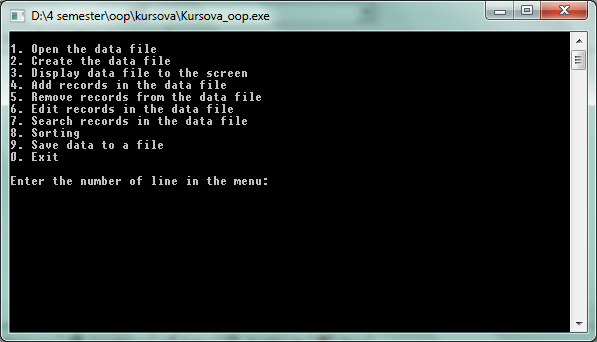


Рис. 5.1 Головне меню програми

Тепер подивимося, що відбувається при натисканні користувачем кожного пункту меню. При невірному введенні даних у всіх пунктах меню виходить повідомлення про помилку. Номери пунктів - номери клавіш на клавіатурі.

1. Відкрити файл даних - в цьому пункті користувач може ввести свій файл.

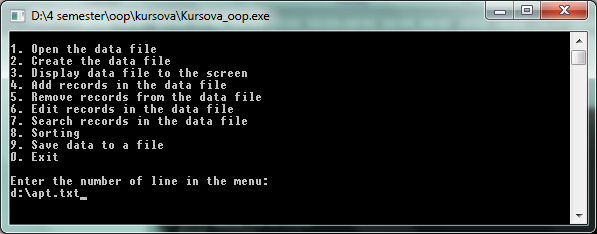
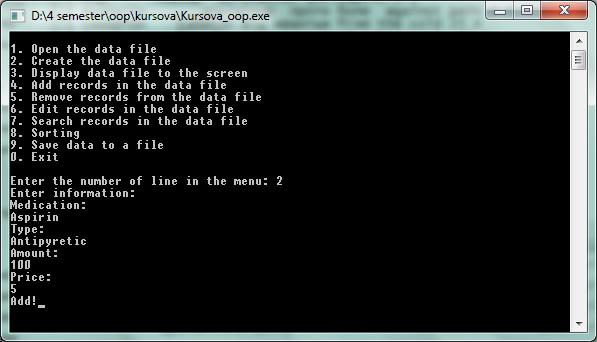


Рис.5.2

2. Створити файл даних - в цьому пункті ми можемо створити новий файл даних.



Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

13

**КР СН-08-033 ПЗ**

Рис. 5.3

3. Вивести файл даних на екран - в цьому пункті користувач може переглянути свій файл.

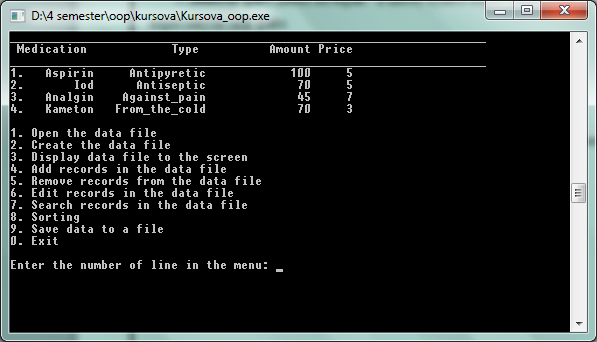


Рис. 5.4

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

14

**КР СН-08-033 ПЗ**

4. Додавання запису у файл даних - в цьому пункті можна додати новий запис.

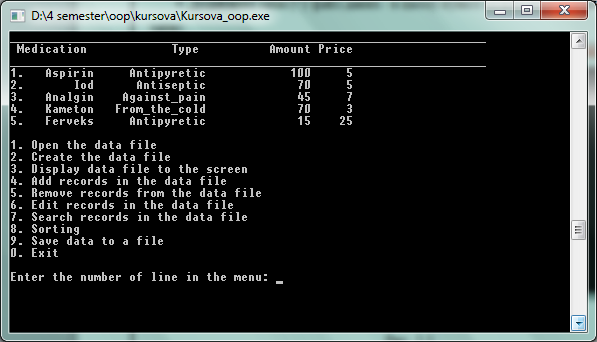
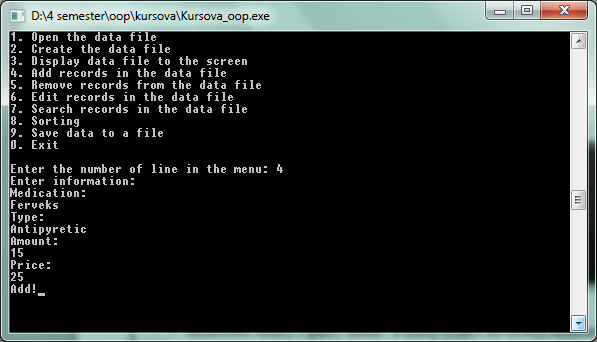


Рис. 5.5

1. Видалення запису з файлу даних - в цьому пункті ви можете видалити будь-який запис.

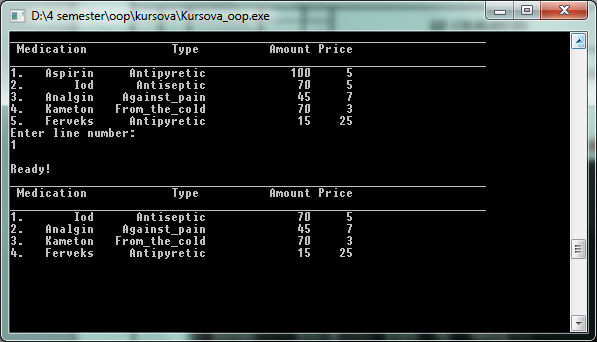


Рис. 5.6

6. Редагування запису у файлі даних - в цьому пункті ви можете відредагувати будь-який запис.

Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

15

**КР СН-08-033 ПЗ**

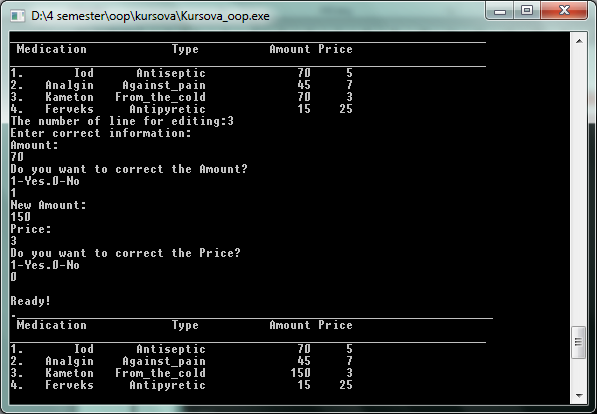


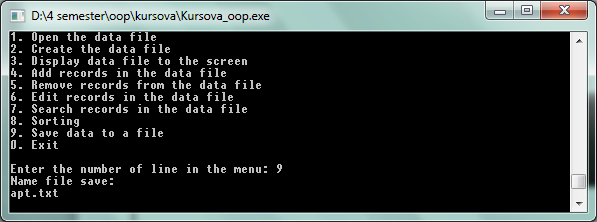
Рис. 5.7.

7. Пошук записів у файлі даних - в цьому пункті ви можете знайти будь-який медикамент за вказаними критеріями



Рис. 5.8

9. Збереження файлу даних у файл - в цьому пункті ви можете зберегти файл даних.



Змн.

Арк.

№ докум.

Підпис

Дата

Арк.

16

**КР СН-08-033 ПЗ**

0. Вихід - при наборі на клавіатурі числа 0 буде здійснений вихід з програми.

**ВИСНОВКИ**

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

17

**КР СН-08-033 ПЗ**

Розроб.

Савчук В.О.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Висновки**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

Отже, результаті виконання даної курсової роботи було розроблено програму для обліку медикаментів в складі лікарні. Програма розроблена в середовищі MicrosoftVisualStudio 2010 . Вона дозволяє додавати, редагувати, видаляти і переглядати дані про наявні на базі ліки. Дана програма має консольний режим взаємодії з користувачем і зберігає результати в файл.

**СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

Змн.

Лист

№ докум.

Підпис

Дата

Арк.

18

**КР СН-08-033 ПЗ**

Розроб.

Савчук В.О.

Перевір.

Бревус В.М.

Реценз.

Н. Контр.

Затверд.

**Список використаної літератури**

Літ.

Акрушів

18

ТНТУ ФІС гр.СН-22

1. Стефенс Д.Р.С++.Сборник рецептов. – КУДИЦ-ПРЕСС, 2007. – 624 с. – ISBN 5-91136-0306

2. К. Джамаса. Учимся програмировать на языке С++: Пер. С англ.. – Москва: Мир, 1997. 320с.

3. Шпак З.Я. Програмування мовою С –Львів: Оріяна-Нова, 2006 р.-432с.

4. С++, Культин Н.Б, СПб:БХВ-Петербург, 2005г. 288с.

5. Герберт Шилдт. Полный справочник по С++=С++:The Complete

Reference. – 4-е изд. – М. Вильямс, 2006. – 800с. ISBN 0-07-2226803

# ДОДАТОК А

## Лістинг Sourse

#include <iostream.h>

#include <conio.h>

#include <string.h>

#include <iomanip.h>

#include <fstream.h>

#include <stdlib.h>

#include “Header.h”

int main()

{

int w;

Pharmacy a;

while(1)

{

cout<<"1. Open the data file\n";

cout<<"2. Create the data file\n";

cout<<"3. Display data file to the screen\n";

cout<<"4. Add records in the data file\n";

cout<<"5. Remove records from the data file\n";

cout<<"6. Edit records in the data file\n";

cout<<"7. Search records in the data file\n";

cout<<"8. Sorting\n";

cout<<"9. Save data to a file\n";

cout<<"0. Exit\n"<<endl;

cout <<"Enter the number of line in the menu: "; cin>>w;

switch(w)

{

case 1:a.reading();break;

case 2:a.create();break;

case 3:a.output();getch();break;

case 4:a.adding();break;

case 5:a.del();break;

case 6:a.correct();break;

case 7:a.find();getch();break;

case 8:a.sorting();break;

case 9:a.record();break;

case 0:exit(0);

default:cout<<"Error";getch(); }

}

}

## Лістинг Header

#include <iostream.h>

#include <conio.h>

#include <string.h>

#include <iomanip.h>

#include <fstream.h>

#include <stdlib.h>

const int L=31, N=100;

struct Apteka

{

char Medication [L];

char Type [L];

int Amount;

int Price;

};

class Pharmacy

{

private:int n;

Apteka \*px;

void Remove(int j);

public:

Pharmacy(){}

~Pharmacy(){if(n>0)delete[]px;}

int k;

void create();

void reading();

void output();

void record();

void find();

void adding();

void del();

void correct();

void sorting();

};

void Pharmacy::create()

{

int i;

px=new Apteka[n];

if(px==NULL)

{

cout<<"No Memory";

getch();

exit(1);

}

n=0;

adding();

do

{

cout<<"To add yet?\n";

cout<<"1-Yes\n";

cout<<"2-No\n";

cin>>i;

if(i==1)adding();

}while(i==1);

}

void Pharmacy::reading()

{

ifstream in;

char file[N];

int i; Apteka u;

cout<<"File name:\n";

cin>>file;

in.open(file);

if(in.fail())

{

cout<<file<<"Don't open";

getch();

exit(1);

}

n=0;

do

{

in>>u.Medication>>u.Type>>u.Amount>>u.Price;

n++;

}while(in.good());

n--;

in.close();

px=new Apteka[n];

if(px==NULL)

{

cout<<"Error: No memory";

getch();

exit(1);}

in.open(file);

if(in.fail())

{

cout<<"Error: "<<file<<" don't open";

getch();

exit(1);}

for(i=0;i<n;i++)

in>>px[i].Medication>>px[i].Type>>px[i].Amount>>px[i].Price;

in.close();

}

void Pharmacy::output()

{

int i;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<" "<<setw(10)<<"Medication"<<" "<<setw(15)<<"Type"<<" "<<setw(15)<<"Amount"<<" "<<setw(5)<<"Price"<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

for(i=0;i<n;i++)

cout<<i+1<<"."<<setw(3)<<setw(10)<<px[i].Medication<<" "<<setw(15)<<px[i].Type<<setw(15)<<px[i].Amount<<" "<<setw(5)<<px[i].Price<<endl;

}

void Pharmacy::find()

{

int b,i,k;

char p [20];

cout<<"Choose a point\n";

cout<<"1)Search to Medication.\n";

cout<<"2)Search to Type.\n";

cout<<"3)Search to Price.\n";

cout<<"Point=\n";

cin>>b;

if(b==1)

{

cout<<"Medication:\n";cin>>p;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<" "<<setw(10)<<"Medication"<<" "<<setw(15)<<"Type"<<" "<<setw(15)<<""<<" "<<setw(5)<<"Amount"<<" "<<setw(10)<<"Price"<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

for(i=0;i<n;i++)

if (strcmp(px[i].Medication,p)==0)

cout<<i+1<<"."<<setw(3)<<setw(10)<<px[i].Medication<<" "<<setw(15)<<px[i].Type<<setw(15)<<px[i].Amount<<" "<<setw(5)<<px[i].Price<<endl;

}

if(b==2)

{

cout<<"Type:\n";cin>>p;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<" "<<setw(10)<<"Medication"<<" "<<setw(15)<<"Type"<<" "<<setw(15)<<"Amount"<<" "<<setw(5)<<"Price"<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

for(i=0;i<n;i++)

if (strcmp(px[i].Type,p)==0)

cout<<i+1<<"."<<setw(3)<<setw(10)<<px[i].Medication<<" "<<setw(15)<<px[i].Type<<setw(15)<<px[i].Amount<<" "<<setw(5)<<px[i].Price<<endl;

}

if(b==3)

{

cout<<"Price:\n";cin>>k;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<" "<<setw(10)<<"Medication"<<" "<<setw(15)<<"Type"<<" "<<setw(15)<<"Amount"<<" "<<setw(5)<<"Price"<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

for(i=0;i<n;i++)

if (px[i].Price==k)

cout<<i+1<<"."<<setw(3)<<setw(10)<<px[i].Medication<<" "<<setw(15)<<px[i].Type<<setw(15)<<px[i].Amount<<" "<<setw(5)<<px[i].Price<<endl;

}

}

void Pharmacy::record()

{

int i;

ofstream out;

char file[N];

cout<<"Name file save:\n";

cin>>file;

out.open(file);

if(out==NULL)

{

cout<<file<<"Don' create";

getch();

exit(1);

}

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

cout<<" "<<setw(10)<<"Medication"<<" "<<setw(15)<<"Type"<<" "<<setw(15)<<"Amount"<<" "<<setw(5)<<"Price"<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

for(i=0;i<n;i++)

out<<i+1<<"."<<setw(3)<<setw(10)<<px[i].Medication<<" "<<setw(15)<<px[i].Type<<setw(15)<<px[i].Amount<<" "<<setw(5)<<px[i].Price<<endl;

getch();out.close();

}

void Pharmacy::adding()

{

Apteka \*s;

int i;

s=new Apteka[n+1];

for(i=0;i<n;i++)

{

s[i]=px[i];

cout<<"Enter information:\n";

cout<<"Medication:\n";cin>>s[n].Medication;

cout<<"Type:\n";cin>>s[n].Type;

cout<<"Amount:\n";cin>>s[n].Amount;

cout<<"Price:\n";cin>>s[n].Price;

}

n++;

delete[]px;

px=s;

cout<<"Add!";

getch();

}

void Pharmacy::correct()

{

int i,q,j;

output();

cout<<"The number of line for editing:";cin>>j;

for(i=0;i<n;i++)

if(i==j-1)

{

cout<<"Enter correct information:\n";

cout<<"Amount:\n"<<""<<px[i].Amount<<endl<<"Do you want to correct the Amount?\n"<<"1-Yes.0-No\n";

cin>>q;

if(q==1){cout<<"New Amount:\n";cin>>px[i].Amount;}

cout<<"Price:\n"<<""<<px[i].Price<<endl<<"Do you want to correct the Price?\n"<<"1-Yes.0-No\n";

cin>>q;

if(q==1){cout<<"New Price:\n";cout<<"";cin>>px[i].Price;

}

cout<<endl<<"Ready!\n.";

break; }

output();

getch();

}

void Pharmacy::del()

{

int i,h,fl;

output();

cout<<"Enter line number:\n";

cin>>h;

fl=0;

for(i=0;i<n;i++)

if(i==h-1)

{

fl=1;Remove(i);

cout<<endl<<"Ready!\n";

}

if(fl==0)cout<<"Not found.\n";

output();

getch();

}

void Pharmacy::Remove(int h)

{

int i;

if(h<0||h>=n)

{

cout<<"Does not exist";

getch();

return;

}

for(i=h+1;i<n;i++)

px[i-1]=px[i];

n--;

}

void Pharmacy::sorting()

{

int i,p,fl;

Apteka r;

cout<<"How sort?\n";

cout<<"---------------\n";

cout<<"1.Sorting Medication\n";

cout<<"2.Sorting Type\n";

cout<<"3.Sorting Amount\n";

cout<<"4.Sorting Price\n";

cout<<"Punkt-";

cin>>p;

do{for(i=0;i<(n-1);i++)

{

if(p==1)

{

fl=0;

if(strcmp(px[i+1].Medication,px[i].Medication)<0)

{

r=px[i];

px[i]=px[i+1];

px[i+1]=r;

fl=1;break;

}

}

if(p==2)

{

fl=0;

if(strcmp(px[i+1].Type,px[i].Type)<0)

{

r=px[i];

px[i]=px[i+1];

px[i+1]=r;

fl=1;break;

}

}

if(p==3)

{

fl=0;

if(px[i+1].Amount<px[i].Amount)

{

r=px[i];

px[i]=px[i+1];

px[i+1]=r;

fl=1;break;

}

}

if(p==4)

{

fl=0;

if(px[i+1].Price<px[i].Price)

{

r=px[i];

px[i]=px[i+1];

px[i+1]=r;

fl=1;break;

}

}

}

}while(fl==1);

}