1. Торгово-развлекательный центр. Электронное хранилище документации.  
2. Туристическое агентство. Клиенты, продажи.  
3. Ресторанный бизнес / общепит. Управление меню.  
4. Медицинские услуги. Врач – истории болезней.  
5. Поставка специализированной техники. Торговля и склад.  
6. Поставка специализированной техники. Контрагенты (клиенты /  
поставщики / субподрядчики на сервисе / кредитные организации).  
7. Гостиничный бизнес. Фонд недвижимого имущества (здания /  
сооружения, номера помещения).  
8. Строительство. Поставщики и подрядчики.  
9. Предприятие оптовой торговли. Обработка заявок на поставку.  
10. Медицинские услуги. Регистратура. Карточки пациентов, запись  
на приём по кабинетам.  
11. Поставка специализированной техники. Сервис.  
12. Ресторанный бизнес / общепит. Бар – торговля и склад.  
13. Строительство. Отдел продаж.  
14. Торгово-развлекательный / офисный центр. Служба работы с  
арендаторами: обработка заявок.  
15. Проектная организация. Управление проектами.  
16. Услуги связи. Техническая поддержка абонентов.  
17. Производство мебели. Торговля и склад, управление индивидуальными заказами и оптовыми поставками.  
18. Медицинские услуги. Инвентаризация оборудования.  
19. Промышленное производство. Электронное хранилище проектной документации.  
20. Строительство. Хранилище проектной документации.  
21. Туристическое агентство. Туры, ценовые предложения.  
22. Автомобильные перевозки. Тарификация и маршруты.  
23. Услуги связи. Подрядчики и поставщики, цепочки поставок.  
24. Санаторий. Фонд недвижимого имущества (здания / сооружения  
/ помещения).  
25. Офисный центр. Служба технического обслуживания.  
26. Промышленное производство. Инвентаризация оборудования  
технологических линий.  
27. Строительство. Парк техники.  
28. Консалтинговая компания. Электронные консультации.  
29. Промышленное производство. Ценовые предложения – формирование прайс-листов.  
30. Ресторанный бизнес / общепит. Обработка заказов: зал / кухня.  
31. Услуги связи. Инвентаризация инфраструктуры.  
32. Предприятие оптовой торговли. Торговля и склад, управление  
запасами, цепочки поставки.  
33. Недвижимость. Электронное хранилище документации.  
34. Промышленное производство. Заказчики и статусы, график поставок, платежи / задолженности.  
35. Автомобильные перевозки. Отдел продаж.  
36. Строительство. Объекты – управление проектами.  
37. Производство мебели. Контрагенты (поставщики / заказчики),  
цепочки поставки.  
38. Туристический оператор. Туры, ценовые предложения.  
39. Торгово-развлекательный / офисный центр. Отдел рекламы кампании / календарь, поставщики.  
40. Медицинские услуги. Регистратура – управление прайс-листом,  
занятость манипуляционных / операционных.  
41. Рекламное агентство. Медиа-планирование (загрузка рекламных  
площадок, очереди контрагентов).  
42. Медицинские услуги. Стационар - палаты, пациенты, курсы лечения, занятость манипуляционных / операционных.  
43. Туристический оператор. Контрагенты (агентства, места размещения, перевозчики).  
44. Торгово-развлекательный / офисный центр. Служба работы с  
арендаторами: карточки, платежи / взносы, история.  
45. Торгово-развлекательный / офисный центр. Инвентаризация  
оборудования.  
46. Ресторанный бизнес. Поставщики и управление запасами.  
47. Проектная организация. Электронное хранилище проектной документации.  
48. Автомобильные перевозки. Парк техники.  
49. Санаторий. Регистратура - карточки отдыхающих, платные  
услуги, запись на приём.  
50. Торгово-развлекательный / офисный центр. Служба безопасности: инциденты, вход / выход в административные помещения.  
51. Торгово-развлекательный. Служба ИТ – обработка заявок.  
52. Туристический оператор. Сделки, бронирование, загруженность  
направлений.  
53. Производство мебели. Модельный ряд и индивидуальная комплектация.  
54. Консалтинговая компания. Управление проектами.  
55. Автосервис. Наряды и оперативное управление.  
56. Промышленное производство. Управление номенклатурой - ценовые предложения, планирование производства.  
57. Рекламное агентство. Карточки клиентов и контрагентов.  
58. Санаторий. Управление запасами и цепочками поставок.  
59. Гостиничный бизнес. Обслуживание номеров, платные услуги,  
работа с заявками.  
60. Торгово-развлекательный / офисный центр. Фонд недвижимого  
имущества (здания / сооружения / помещения).  
61. Промышленное производство. Управление запасами и цепочками поставок, поставщики / субподрядчики.  
62. Гостиничный бизнес. Ресепшн, номерной фонд, клиенты.  
63. Проектная организация. Служба поддержки заказчиков.  
64. Недвижимость. Клиенты / агенты / сделки.  
65. Медицинские услуги. Регистратура – расписание приёма врачей,  
работы лабораторий, запись на приём.  
66. Инвестиционные компании. Управление инвестиционными  
портфелями.

Завдання 3

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <iostream>

#include<Windows.h>

#include<math.h>

#include<string.h>

#include<stdlib.h>

#include<direct.h>

#include<io.h>

#include<time.h>

using namespace std;

#define size 100

struct Menu

{

char name[size];

int cost;

int massa;

int kkal;

int time;

char kitchen[size];

};

void Print\_Menu(Menu\* menu, int s)

{

for (int i = 0; i < s; i++)

{

cout << menu[i].name << " " << menu[i].cost << "грн " << menu[i].massa << "грам " << menu[i].kkal << "kkal " << menu[i].time << "хв Кухня - " << menu[i].kitchen << endl;

}

}

Menu\* Add\_Strava(Menu\* arr, int\* s)

{

Menu\* temp = new Menu[++(\*s)];

for (int i = 0; i < \*s - 1; i++)

{

strcpy(temp[i].name, arr[i].name);

temp[i].cost = arr[i].cost;

temp[i].massa = arr[i].massa;

temp[i].kkal = arr[i].kkal;

temp[i].time = arr[i].time;

strcpy(temp[i].kitchen, arr[i].kitchen);

}

cout << "Введіть назву страви ->";

cin >> temp[\*s - 1].name;

cout << "Введіть ціну страви ->";

cin >> temp[\*s - 1].cost;

cout << "Введіть масу страви ->";

cin >> temp[\*s - 1].massa;

cout << "Введіть кількість кілокалорій ->";

cin >> temp[\*s - 1].kkal;

cout << "Введіть час приготування ->";

cin >> temp[\*s - 1].time;

cout << "Введіть кухню ->";

cin >> temp[\*s - 1].kitchen;

delete[] arr;

return temp;

}

void Zamovlenna(Menu\* menu, int s)

{

bool b = false;

int cost2=0;

char name2[size];

cout << "Введіть страву (Закінчити замовляти 0)->";

cin >> name2;

while (strcmp(name2, "0") != 0)

{

for (int i = 0; i < s; i++)

{

if (strcmp(menu[i].name, name2) == 0)

{

cost2 += menu[i].cost;

b = true;

break;

}

}

if (!b)

cout << "Немає такої страви!\n";

b = false;

cout << "Ще замовити?Якщо так то введіть назву страви (Закінчити замовляти 0)->";

cin >> name2;

}

cout << "Ціна замовлення - " << cost2 << "грн\n";

}

Menu\* Delete\_Strava(Menu\* menu, int\* s)

{

Menu\* temp = new Menu[--(\*s)];

int index;

char name[size];

bool b = false;

cout << "Введіть назву страви ->";

cin >> name;

for (int i = 0; i <= \*s; i++)

{

if (strcmp(name, menu[i].name)==0)

{

index = i;

b = true;

}

}

if (b)

{

for (int i = 0; i < index; i++)

{

strcpy(temp[i].name, menu[i].name);

temp[i].cost = menu[i].cost;

temp[i].massa = menu[i].massa;

temp[i].kkal = menu[i].kkal;

temp[i].time = menu[i].time;

strcpy(temp[i].kitchen, menu[i].kitchen);

}

int k = index;

for (int i = index + 1; i < \*s + 1; i++)

{

strcpy(temp[k].name, menu[i].name);

temp[k].cost = menu[i].cost;

temp[k].massa = menu[i].massa;

temp[k].kkal = menu[i].kkal;

temp[k].time = menu[i].time;

strcpy(temp[k].kitchen, menu[i].kitchen);

k++;

}

delete[] menu;

cout << "Успішно видалено!\n";

return temp;

}

else

{

++(\*s);

delete[] temp;

cout << "Страву не знайдено!\n";

return menu;

}

}

Menu\* Redact\_Cost(Menu\* menu, int s)

{

char name[size];

bool b = false;

cout << "Введіть назву страви ->";

cin >> name;

for (int i = 0; i <= s; i++)

{

if (strcmp(name, menu[i].name) == 0)

{

cout << "Введіть нову ціну ->";

cin >> menu[i].cost;

b = true;

return menu;

}

}

if(!b)

{

cout << "Страву не знайдено!\n";

return menu;

}

}

Menu\* Redact\_Massa(Menu\* menu, int s)

{

char name[size];

bool b = false;

cout << "Введіть назву страви ->";

cin >> name;

for (int i = 0; i <= s; i++)

{

if (strcmp(name, menu[i].name) == 0)

{

cout << "Введіть нову масу ->";

cin >> menu[i].massa;

b = true;

return menu;

}

}

if (!b)

{

cout << "Страву не знайдено!\n";

return menu;

}

}

Menu\* Sort\_by\_cost(Menu\* menu, int\* s)

{

bool b;

do {

b = false;

for (int i = 0; i < \*s - 1; i++) {

if (menu[i].cost> menu[i+1].cost)

{

char temp[size];

int temp2;

strcpy(temp, menu[i].name);

strcpy(menu[i].name, menu[i+1].name);

strcpy(menu[i + 1].name, temp);

temp2 = menu[i].cost;

menu[i].cost = menu[i + 1].cost;

menu[i + 1].cost = temp2;

temp2 = menu[i].massa;

menu[i].massa = menu[i+1].massa;

menu[i + 1].massa = temp2;

temp2 = menu[i].kkal;

menu[i].kkal = menu[i+1].kkal;

menu[i + 1].kkal = temp2;

temp2 = menu[i].time;

menu[i].time = menu[i+1].time;

menu[i + 1].time = temp2;

strcpy(temp, menu[i].kitchen);

strcpy(menu[i].kitchen, menu[i+1].kitchen);

strcpy(menu[i + 1].kitchen, temp);

b = true;

}

}

} while (b);

cout << "Відсортовано!" << endl;

return menu;

}

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int s = 3, k, a2=-1;

FILE\* file;

Menu\* menu = new Menu[s];

char str[size \* 4];

char name[size];

cout << "Введіть шлях і назву файла -> "; //E:\TEST\test

cin >> name;

strcat(name, ".txt");

file = fopen(name, "w");

fprintf(file, "Borshch 120 200 250 30 Ukraine\nGolybtsi 200 150 300 45 Ukraine\nSushi 500 300 600 55 Japan");

fclose(file);

file = fopen(name, "r");

char separator[] = "\n ";

char\* pstr;

for (int i = 0; i < s; i++)

{

fgets(str, size \* 4, file);

pstr = strtok(str, separator);

strcpy(menu[i].name, pstr);

pstr = strtok(NULL, separator);

menu[i].cost = atoi(pstr);

pstr = strtok(NULL, separator);

menu[i].massa = atoi(pstr);

pstr = strtok(NULL, separator);

menu[i].kkal = atoi(pstr);

pstr = strtok(NULL, separator);

menu[i].time = atoi(pstr);

pstr = strtok(NULL, separator);

strcpy(menu[i].kitchen, pstr);

}

fclose(file);

int a;

cout << "Режим гостя - 1, режим адміністратора - 2\n";

cout << "->";

cin >> a;

while (a2!=0)

{

switch (a)

{

case 1:

cout << "Що бажаєте замовити?\n";

Print\_Menu(menu, s);

Zamovlenna(menu, s);

break;

case 2:

cout << "Введіть Пін-код\n";

cout << "->";

cin >> k;

if (k == 1111)

{

cout << "Додати страву - 1, видалити страву - 2, змінити ціну страви - 3, змінити масу страви - 4, переглянути меню - 5, сортування по ціні - 6, вихід до попереднього меню - -1, виключити систему - 0\n";

cout << "->";

cin >> a2;

while (a2 != -1 && a2 != 0)

{

switch (a2)

{

case 1:

menu = Add\_Strava(menu, &s);

break;

case 2:

menu = Delete\_Strava(menu, &s);

break;

case 3:

menu = Redact\_Cost(menu, s);

break;

case 4:

menu = Redact\_Massa(menu, s);

break;

case 5:

Print\_Menu(menu, s);

break;

case 6:

menu = Sort\_by\_cost(menu, &s);

break;

}

cout << "Додати страву - 1, видалити страву - 2, змінити ціну страви - 3, змінити масу страви - 4, переглянути меню - 5, сортування по ціні - 6, вихід до попереднього меню - -1, виключити систему - 0\n";

cout << "->";

cin >> a2;

}

break;

}

else

{

cout << "Не правильний пін-код!\n";

break;

}

}

if (a2 != 0)

{

cout << "Режим гостя - 1, режим адміністратора - 2\n";

cout << "->";

cin >> a;

}

}

file = fopen(name, "w");

for (int i = 0; i < s; i++)

{

fprintf(file, "%s %i %i %i %i %s\n", menu[i].name, menu[i].cost, menu[i].massa, menu[i].kkal, menu[i].time, menu[i].kitchen);

}

fclose(file);

delete[] menu;

return 0;

}