ЛАБОРАТОРНАЯ РАБОТА № 3

ИССЛЕДОВАНИЕ МЕХАНИЗМА МНОЖЕСТВЕННОГО НАСЛЕДОВАНИЯ

1 ЦЕЛЬ РАБОТЫ

Приобретение практических навыков при написании объектно-ориентированных программ с использованием механизма множественного наследования.

2 ПОСТАНОВКА ЗАДАЧИ

Описать интерфейс ввода-вывода. Описать иерархию классов, заданную по варианту. Для каждого класса описать конструкторы и деструктор (по необходимости), функции ввода и вывода значений полей. В каждом классе должны присутствовать минимум одно уникальное поле и один уникальный метод. Проиллюстрировать корректную работу механизма множественного наследования — для этого создать объекты базовых классов и заполнить их поля данными, вывести на печать. Создать объект класса-наследника, его поля заполнить значениями соответствующих полей базовых классов. Вывести на печать данные полученного объекта.

Вариант 3

Базовые классы: Устройство (название, мощность, производитель), Звук (частота, громкость). Класс-наследник: Сирена.

3 ТЕКСТ ПРОГРАММЫ

#include <conio.h>

#include <stdlib.h>

#include <iostream>

using namespace std;

class Device

{

public:

char name[15], prod[20];

float pow;

Device()

{

cout<<"Название: ";

cin>>name;

cout<<"Производитель: ";

cin>>prod;

cout<<"Мощность: ";

cin>>pow;

}

void Output()

{

cout<<"Перечень устройств:"<<endl<<endl;

cout<<"Название: "<<name<<endl;

cout<<"Производитель: "<<prod<<endl;

cout<<"Мощность: "<<pow<<endl;

}

};

class Tune

{

public:

float freq, vol;

Tune()

{

cout<<"Базовая громкость(дБ): ";

cin>>vol;

cout<<"Базовая частота (Гц): ";

cin>>freq;

}

void Output()

{

cout<<"Необходимые параметры:"<<endl;

cout<<"Громкость: "<<vol<<endl;

cout<<"Частота: "<<freq<<endl;

}

};

class Siren : public Device, public Tune

{

public:

int r1, r2;

void Range()

{

cout<<"Добавьте необходимый диапазон частот: "<<endl;

cout<<"Нижний порог: ";

cin>>r1;

cout<<"Верхний порог: ";

cin>>r2;

}

void Output()

{

cout<<"Название: " <<name<<endl;

cout<<"Производитель: "<<prod<<endl;

cout<<"Громкость: "<<vol<<endl;

cout<<"Диапазон частот: "<<r1<<".."<<r2<<endl;

}

};

int main ()

{

setlocale(0, "RUS");

int k;

Siren ob;

while (1)

{

system("cls");

cout << "1 - Вызов объектом производного класса методов базовых классов(Просмотр)" << endl;

cout << "2 - Просмотр объекта производного класса " << endl;

cout << "3 - Вызов уникального метода производного класса " << endl;

cout << "4 - Выход " << endl;

k=getch();

switch (k)

{

case'1':

{

ob.Device::Output();

cout<<"-----------------------"<<endl;

ob.Tune::Output();

system("pause");

break;

}

case '2':

{

cout<<"Данные устройства:"<<endl;

ob.Output();

system("pause");

break;

}

case '3':

{

ob.Range();

system("pause");

break;

}

case '4':

return 0;

default:

cout <<"Выберите пункт меню"<< endl;

}

}

return 0;

}

4 ТЕСТИРОВАНИЕ

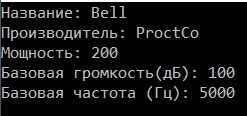


Рисунок 1 – Ввод параметров базовых классов

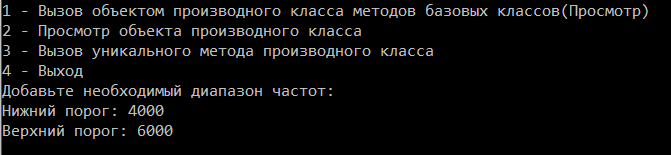


Рисунок 2 – Тестирование работы функции производного класса

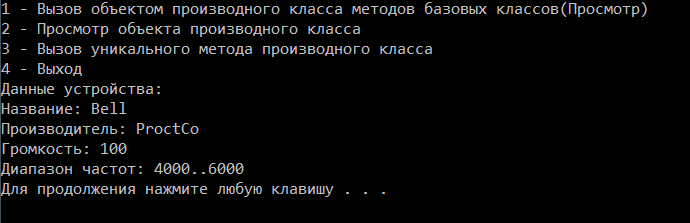


Рисунок 3 – Просмотр объекта производного класса

ВЫВОД

В ходе выполнения лабораторной работы были приобретены практические навыки при написании объектно-ориентированных программ с использованием механизма множественного наследования.