**ЛАБОРАТОРНАЯ РАБОТА №6**

**НАСЛЕДОВАНИЕ**

**Цель**:

Целью данной лабораторной работы является изучение наследованная классов в языке С++.

**Задания:**

1) Создать класс «Староста», производный от класса «Студент». Новый класс должен содержать несколько дополнительных методов и полей.

2) Создать класс Alive и расширить его до Bird, Fish, Animal

3) Создать класс Animal, и расширить его до Dog, Cat.

**Задание №1**

#include "pch.h"

#include <iostream>

#include "windows.h"

#include<string>

using namespace std;

class Student

{

protected:

string name;

string surname;

string group;

public:

virtual void set()

{

cout << "Введите имя студента:" << endl;

cin >> name;

cout << "Введите фамилию студента:" << endl;

cin >> surname;

cout << "Введите группу студента:" << endl;

cin >> group;

}

virtual void get()

{

cout << "Меня зовут " <<surname<<" "<<name<<" я студент группы "<<group<< endl;

}

};

class Headman : public Student

{

protected:

string email;

string phone;

public:

void set()

{

cout << "Введите имя старосты:" << endl;

cin >> name;

cout << "Введите фамилию старосты:" << endl;

cin >> surname;

cout << "Введите группу старосты:" << endl;

cin >> name;

cout << "Введите email старосты:" << endl;

cin >> email;

cout << "Введите телефон старосты:" << endl;

cin >> phone;

}

void get()

{

cout << "Меня зовут " << surname << " " << name << " я староста группы " << group << endl;

cout << "Мой email:"<< email << endl;

cout << "Мой телефон:" << phone << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

Student a;

Headman b;

a.set();

a.get();

b.set();

b.get();

system("pause");

return 0;

}

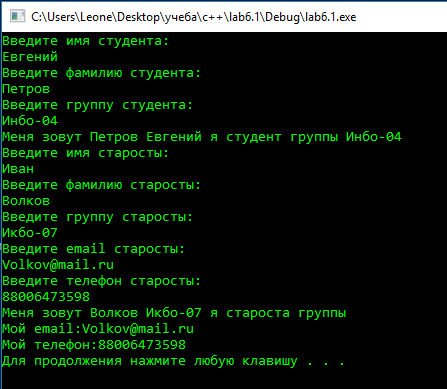
****

Рис.1

**Задание №2**

#include "pch.h"

#include <iostream>

#include<string>

#include "windows.h"

using namespace std;

class Alive

{

protected:

string name;

int year;

public:

void set()

{

cout << "Введите имя:";

cin >> name;

cout << "Введите возраст:";

cin >> year;

}

void out()

{

cout << "Я " << name << " мне " << year << endl;

}

virtual void act() = 0;

};

class Animal : public Alive

{

public:

void act()

{

cout << "Я бегу" << endl;

}

};

class Bird : public Alive

{

public:

void act()

{

cout << "Я лечу" << endl;

}

};

class Fish : public Alive

{

public:

void act()

{

cout << "Я плыву" << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

Bird a;

cout << "Птица" << endl;

a.set();

a.out();

a.act();

cout << endl;

Fish b;

cout << "Рыба" << endl;

b.set();

b.out();

b.act();

cout << endl;

Animal c;

cout << "Животное" << endl;

c.set();

c.out();

c.act();

system("pause");

return 0;

}

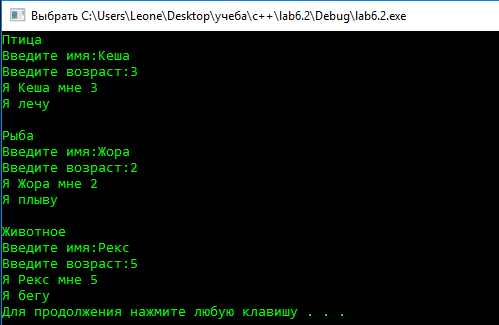
****

Рис.2

**Задание №3**

#include "pch.h"

#include <iostream>

#include<string>

#include "windows.h"

using namespace std;

class Animal

{

protected:

string name;

int year;

public:

void set()

{

cout << "Введите имя:";

cin >> name;

cout << "Введите возраст:";

cin >> year;

}

void out()

{

cout << "Я " << name << " мне " << year << endl;

}

virtual void act() = 0;

};

class Dog : public Animal

{

public:

void act()

{

cout << "Гав Гав Гав!!!" << endl;

}

};

class Cat : public Animal

{

public:

void act()

{

cout << "Мяяяу!!!" << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

Dog a;

cout << "Cобака" << endl;

a.set();

a.out();

a.act();

cout << endl;

Cat b;

cout << "Кот" << endl;

b.set();

b.out();

b.act();

cout << endl;

system("pause");

return 0;

}

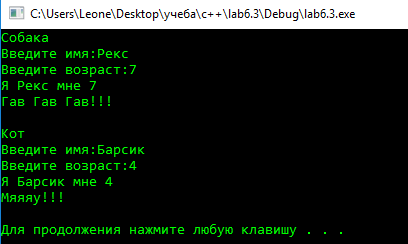


Рис.3

**Вывод:** В ходе выполнения лабораторной работы №6 было реализовано наследование классов, а также использована виртуальная функция.