**Практическая работа №7**

**АБСТРАКТНЫЕ КЛАССЫ. МНОЖЕСТВЕННОЕ НАСЛЕДОВАНИЕ**

**Цель:** Целью данной практической работы является знакомство с абстрактными классами и механизмом виртуальных функций, а также множественным наследованием в языке программирования С++.

**Задания:**

1) Реализовать абстрактный класс «Животное» и путём наследования от него получить классы «Кошка», «Собака», «Попугай».

2) Реализовать абстрактный класс «Фигура» и путём наследования от него получить абстрактный класс «Четырёхугольник», и затем путём наследования получить классы «Ромб», «Прямоугольник».

3) Реализовать абстрактный класс «Транспортное средство» и путём наследования от него получить классы «Автомобиль», «Автобус», «Велосипед».

4) Реализовать абстрактные классы «Экран» и «Клавиатура», путём наследования от них получить классы «Ноутбук», Телефон», «Стационарный компьютер».

**Задание №1**

#include "pch.h"

#include <iostream>

#include<string>

#include "windows.h"

using namespace std;

class Animal//абстрактный базовый класс животное

{

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 golos() = 0;

};

class Cat : public Animal //производный класс кошка

{

public:

void golos()

{

cout << "МЯУ!!!" << endl;

}

};

class Dog : public Animal//производный класс собака

{

public:

void golos()

{

cout << "ГАВ ГАВ ГАВ!!!" << endl;

}

};

class Parrot : public Animal//производный класс попугай

{

public:

void golos()

{

cout << "ПРРРИВЕТ!!!" << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

Cat a;

cout << "кот" << endl;

a.set();

a.out();

a.golos();

cout << endl;

Dog b;

cout << "собака" << endl;

b.set();

b.out();

b.golos();

cout << endl;

Parrot c;

cout << "попугай" << endl;

c.set();

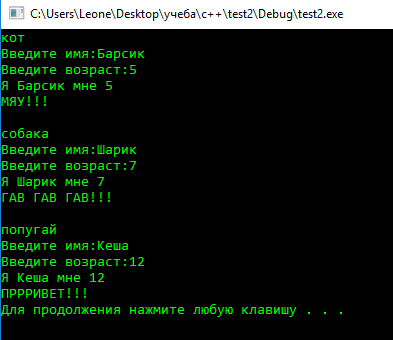
c.out();

c.golos();

system("pause");

return 0;

}



**Задание №2**

#include "pch.h"

#include <iostream>

#include<string>

#include "windows.h"

using namespace std;

class Figure //абстрактный базовый класс фигура

{

protected:

double sqr;

public:

virtual void square() = 0;

};

class Quadrilateral : public Figure //производный абстрактный класс четырехугольник

{

protected:

double a1, a2, a3, a4;

public:

void enter()

{

cout << "Введите длину двух сторон четырехуголника:";

cin >> a1 >> a2 ;

}

virtual void square() = 0;

};

class Rhombus : public Quadrilateral //производный класс ромб

{

private:

double h;

public:

void set()

{

cout << "Введите высоту ромба:" ;

cin >> h;

}

void square()

{

sqr = a1 \* h;

cout << "Площадь ромба="<< sqr<<endl;

}

};

class Rect : public Quadrilateral //производный класс прямоугольник

{

public:

void square()

{

sqr = a1 \* a2;

cout << "Площадь прямоугольника=" << sqr << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

Rhombus a;

a.enter();

a.set();

a.square();

Rect b;

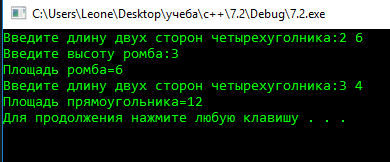
b.enter();

b.square();

system("pause");

return 0;

}



**Задание №3**

#include "pch.h"

#include <iostream>

#include<string>

#include "windows.h"

using namespace std;

class Vehicle//абстрактный базовый класс транспортное средство

{

protected:

string model;

string name;

int speed;

public:

void set()

{

cout << "Введите марку:";

cin >> name;

cout << "Введите модель:";

cin >> model;

cout << "Введите максимальную скорость:";

cin >> speed;

}

virtual void out() = 0;

};

class avto : public Vehicle //производный класс автомобиль

{

public:

void out()

{

cout << "максимальная скорость автомобиля " << name << " "<< model <<" "<< speed<< "км/ч" << endl;

}

};

class Bycicle : public Vehicle//производный класс велосипед

{

public:

void out()

{

cout << "максимальная скорость велосипеда " << name << " " << model << " " << speed << "км/ч" << endl;

}

};

class Bus : public Vehicle//производный класс автобус

{

public:

void out()

{

cout << "максимальная скорость автобуса " << name << " " << model << " " << speed << "км/ч" << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

cout << "авто" << endl;

avto a;

a.set();

a.out();

cout << endl;

cout << "велосипед" << endl;

Bycicle b;

b.set();

b.out();

cout << endl;

cout << "автобус" << endl;

Bus c;

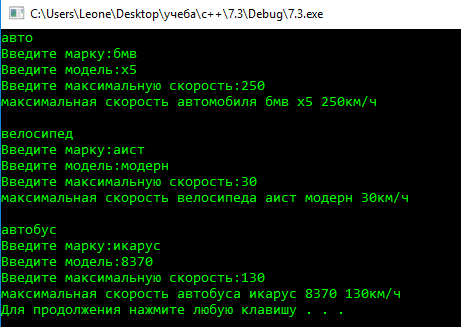
c.set();

c.out();

system("pause");

return 0;

}

****

**Задание №4**

#include "pch.h"

#include <iostream>

#include<string>

#include "windows.h"

using namespace std;

class Screen //абстрактный базовый класс экран

{

protected:

int wide;

int length;

public:

virtual void resolution() = 0;

};

class Keyboard //абстрактный базовый клавиатура

{

protected:

string type;

public:

virtual void klava() = 0;

};

class Laptop: public Screen, Keyboard //производный абстрактный класс ноутбук

{

public:

void resolution()

{

cout << "Введите разрешение по ширине и длине:";

cin >> wide >> length;

}

void klava()

{

cout << "Введите тип клавиатуры:";

cin >> type;

}

void out()

{

cout << "Разрешение экрана ноутбука "<<wide<<"x"<<length<<endl;

cout << "Тип клавиатуры ноутбука " << type<< endl;

}

};

class Phone : public Screen, Keyboard //производный класс телефон

{

public:

void resolution()

{

cout << "Введите разрешение по ширине и длине:";

cin >> wide >> length;

}

void klava()

{

cout << "Введите тип клавиатуры:";

cin >> type;

}

void out()

{

cout << "Разрешение экрана телефона " << wide << "x" << length << endl;

cout << "Тип клавиатуры телефона " << type << endl;

}

};

class PC : public Screen, Keyboard //производный класс ПК

{

public:

void resolution()

{

cout << "Введите разрешение по ширине и длине:";

cin >> wide >> length;

}

void klava()

{

cout << "Введите тип клавиатуры:";

cin >> type;

}

void out()

{

cout << "Разрешение экрана ПК " << wide << "x" << length << endl;

cout << "Тип клавиатуры ПК " << type << endl;

}

};

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

system("color 0A");

cout << "Ноутбук" << endl;

Laptop a;

a.resolution();

a.klava();

a.out();

cout << "Телефон" << endl;

Phone b;

b.resolution();

b.klava();

b.out();

cout << "Пк" << endl;

PC c;

c.resolution();

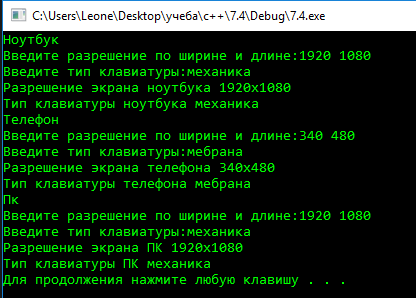
c.klava();

c.out();

system("pause");

return 0;

}



**Вывод:** В ходе выполнения практической работы №7 были реализованы абстрактные классы и продемонстрированы виртуальные функции.