Федеральное государственное образовательное бюджетное учреждение

высшего образования «Финансовый университет при Правительстве

Российской Федерации»

Отчёт по дисциплине «Учебная Практика»

Практическая работа №6

Выполнил студент

группы 2ПКС-116

Зайцев Н. В.

Москва 2018

**Вариант 10:**

1) Создать базовый класс травы и производные классы петрушка и укроп

2)

**Вариант 4**

Написать программу, в которой создаются и разрушаются объекты класса, определенного пользователем. Вывести на печать вызовы конструкторов и деструкторов. Для заполнения полей использовать конструктор без параметров и конструктор с параметрами.Создать массив объектов класса точка (item) из 3 экземпляров c полями x,y и класс 1 круг(krug) c радиусом r . Определить, выходит ли точка за круг .Для обращения к закрытым полям можно использовать дружественную функцию.

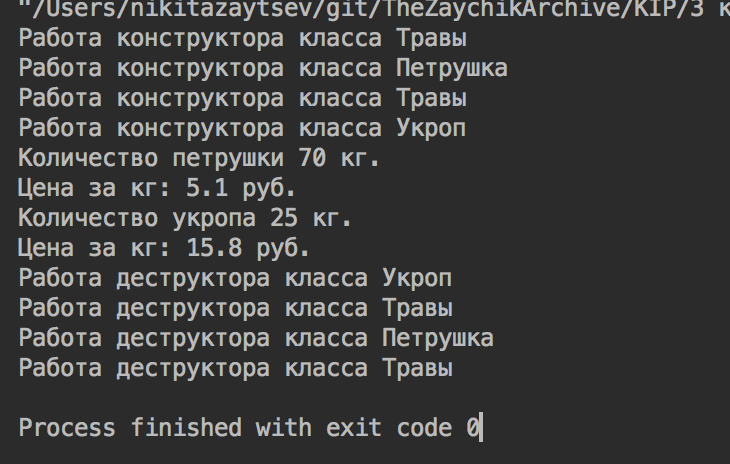
**Код к Заданию 1:**

#include <iostream>  
#include <math.h>  
using namespace std;  
  
class travi{  
protected:  
 double kolvo;  
 double price;  
 travi(double h, double w){  
 cout << "Работа конструктора класса Травы" << endl;  
 this->kolvo = h;  
 this->price = w;  
 }  
 ~travi(){  
 cout << "Работа деструктора класса Травы" << endl;  
 }  
  
 void show();  
};  
class petrushka : public travi{  
public:  
 petrushka(double h, double w) : travi(h,w){  
 cout << "Работа конструктора класса Петрушка" << endl;  
 }  
 void show(){  
 cout << "Количество петрушки " << this->kolvo << " кг." << endl;  
 cout << "Цена за кг: " << this->price << " руб." << endl;  
 }  
 ~petrushka(){  
 cout << "Работа деструктора класса Петрушка" << endl;  
 }  
};  
  
class ukrop : public travi{  
public:  
 ukrop(double h, double w) : travi(h,w){  
 cout << "Работа конструктора класса Укроп" << endl;  
 }  
 void show(){  
 cout << "Количество укропа " << this->kolvo << " кг." << endl;  
 cout << "Цена за кг: " << this->price << " руб." << endl;  
 }  
 ~ukrop(){  
 cout << "Работа деструктора класса Укроп" << endl;  
 }  
};  
  
  
int main(){  
 petrushka r1(70,5.1);  
 ukrop i1(25,15.8);  
 r1.show();  
 i1.show();  
 return 0;  
}

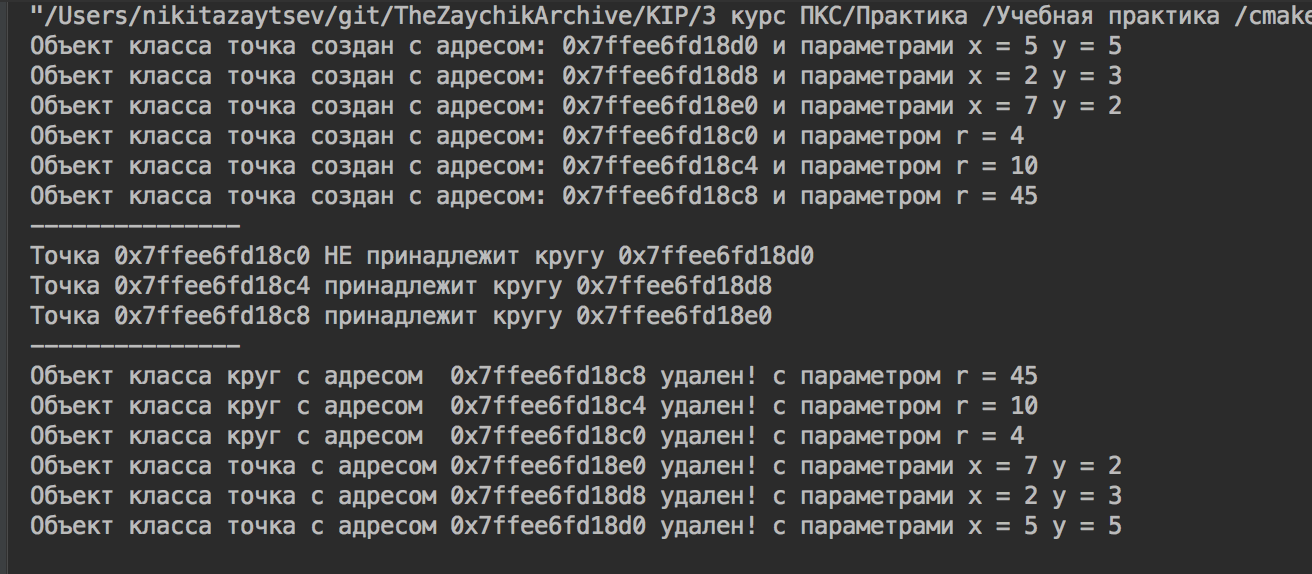
**Код к Заданию 2:**

#include <iostream>  
#include <math.h>  
using namespace std;  
// var 2  
class circle{  
private:  
 friend class dot;  
 int r;  
public:  
 circle(int \_r){  
 this->r = \_r;  
 cout << "Объект класса точка создан с адресом: " << this;  
 cout << " и параметром r = " << this->r << endl;  
 }  
 ~circle(){  
 cout << "Объект класса круг с адресом " << this << " удален! ";  
 cout << "с параметром r = " << this->r << endl;  
 }  
};  
  
class dot{  
private:  
 int x, y;  
public:  
 dot(int \_x, int \_y){  
 this->x = \_x;  
 this->y = \_y;  
 cout << "Объект класса точка создан с адресом: " << this;  
 cout << " и параметрами x = " << this->x << " y = " << this->y << endl;  
 }  
  
 ~dot(){  
 cout << "Объект класса точка с адресом " << this << " удален!";  
 cout << " c параметрами x = " << this->x << " y = " << this->y << endl;  
 }  
  
 void isBelongs(circle &obj){  
 if((x\*x + y\*y) <= obj.r \* obj.r){  
 cout << "Точка " << &obj << " принадлежит кругу " << this << endl;  
 } else{  
 cout << "Точка " << &obj << " НЕ принадлежит кругу " << this << endl;  
 }  
 }  
};  
  
int main(){  
 dot d[3] = {dot(5,5),dot(2,3), dot(7,2)};  
 circle c[3] = {4,10,45};  
 cout << "---------------" << endl;  
 for (int i = 0; i < 3; i++){  
 d[i].isBelongs(c[i]);  
 }  
 cout << "---------------" << endl;  
 return 0;  
}

**Скриншоты к Заданию 1:**

****

**Скриншоты к Заданию 2:**

****