**Покидько Максим Сергеевич**

Дисциплина

Pro

Группа

РC++ш-ЦПД-242211/30-11МОо

**Домашнее задание Повторение указателей и классов. Знакомство с Qt.**

**Задание 1**

// 1. Твоё задание - описать Утконоса Перри и Огурчика Рика с помощью классов. Подумай, какие поля и методы могут помочь

// тебе в этой задаче (например, Перри может выяснять, преступник заданный человек или нет, а Рик может считать, сколько крыс

// он победил) и как с помощью механизма наследования, ты можешь упростить её себе. Реализуй минимум 2 класса и используй

// наследование (минимум для одного класса). Объект каждого класса как минимум должен содержать информацию об имени персонажа

// и мультике откуда он. Также создай 1-2 метода для каждого класса. Можешь использовать переопределение, виртуальные и

// статические методы - всё на твой вкус. Не забудь протестировать работу классов в функции main().

#include <iostream>

/// <summary>Персонажи</summary>

class Character

{

public:

std::string name;

std::string cartoon;

Character(std::string name = "anonymous", std::string cartoon = "unknown") {

this->name = name;

this->cartoon = cartoon;

}

// наследуемый метод

void hello() {

std::cout << "\nПривет, я " << name << "\n";

}

// статический метод

static void message() {

std::cout << "Программа про утконосов и огурцы.\n";

}

// виртуальный метод

virtual void source() {}

};

/// <summary>Утконосы</summary>

class Platypus : public Character

{

public:

Platypus(std::string name, std::string cartoon) : Character(name, cartoon) {}

// реализация виртуального метода

void source() {

std::cout << "Я из мультсериала \"" << cartoon << "\" и могу определять преступников:\n";

}

// выяснять, преступник заданный человек или нет

void report() {

if (std::rand() % 2 == 0)

std::cout << "Преступник\n";

else

std::cout << "Не преступник\n";

}

};

/// <summary>Огурцы</summary>

class Pickle : public Character

{

public:

int frags = 0;

Pickle(std::string name, std::string cartoon) : Character(name, cartoon) {}

// переопределенный метод

void hello() {

std::cout << "\nПРИВЕТИЩЕ, я " << name << "\n";

}

// реализация виртуального метода

void source() {

std::cout << "А я из мультфильма \"" << cartoon << "\" и могу побеждать крыс:\n";

}

// считать, сколько крыс победил

void report() {

this->frags++;

std::cout << "Победы над крысами: " << frags << std::endl;

}

};

int main() {

setlocale(LC\_ALL, "Russian");

srand(time(NULL));

Character::message();

Platypus Perry = Platypus("Perry", "Phineas and Ferb");

Perry.hello();

Perry.source();

Perry.report();

Perry.report();

Perry.report();

Pickle Rick = Pickle("Rick", "Rick and Morty");

Rick.hello();

Rick.source();

Rick.report();

Rick.report();

Rick.report();

return 0;

}

