

УДК 519.682

ОБЪЕКТНО-ОРИЕНТИРОВАННЫЕ ПРОЕКТЫ С СОЗДАНИЕМ КЛАССОВ ДЛЯ МАТЕМАТИЧЕСКИХ ОБЪЕКТОВ

О.А. Широкова

Казанский (Приволжский) федеральный университет, Казань

shirokova2602@mail.ru

Аннотация

Рассмотрены современные проблемы технологии разработки программного обеспечения, основанного на объектно-ориентированном подходе в области программирования.

Ключевые слова: объектно-ориентированное программирование, класс, объект

Объектно-ориентированное программирование (ООП) является дисциплиной, входящей в базовую часть учебных программ для студентов математического факультета. Обучение объектно-ориентированному программированию имеет свои особенности, связанные с высокой степенью абстракции и теоретической сложностью, требующей серьезной теоретической подготовки.

Применение объектно-ориентированного подхода формирует у обучающихся объектное мышление. Однако при этом возникает ряд проблем. Эти проблемы связаны со сложностью изучаемой предметной области, неумением выделить в ней необходимые классы и объекты, их связи и структуры.

В настоящее время объектно-ориентированный стиль применяется при разработке широкого круга приложений [1–8]. Способность студентов мыслить объектно формируется при разработке визуальных приложений с использованием стандартных объектов (компонентов) системы объектно-ориентированного программирования C++. При изучении этого курса студентам предлагается разработка визуальных проектов [3, 6, 7]. Студентов нужно обучать применению знаний в реальных ситуациях, расширять сферу возможного применения ООП. Для этого рекомендуется решать задачи, имеющие объекты, прототипами которых являются реально существующие математические объекты и структуры.

Основной задачей при разработке объектно-ориентированного проекта является программирование с использованием библиотеки визуальных компонентов. При этом обучающийся должен знать иерархию компонентов, которая описывает их взаимодействие; окружение программы, в котором работают компоненты; взаимодействие программы с операционной системой.

Возможности ООП можно эффективно использовать при реализации алгоритмов вычислительного типа, например, при нахождении площади треугольника, заданного своими координатами в пространстве.

Рассмотрим следующий проект: ввести класс точек в пространстве, содержащий как поля – цилиндрические координаты, как методы – функции, определяющие расстояние этой точки до другой точки и расстояние точки до произвольной прямой. Найти площадь треугольника, заданного координатами в пространстве.

Класс CylinderCoord содержит поля – цилиндрические координаты точки. Имеет два конструктора: по умолчанию (все координаты нулевые) и с параметрами (принимает фактические координаты точки). Открытые методы класса позволяют находить расстояние от данной точки до произвольной точки или прямой. Также переопределён оператор `=`, позволяющий сравнивать две точки. Получение данных от пользователя производится через интерфейс Windows Forms. Приведем описание структуры классов и методов в C++.

Source.cpp:

```
#include "MyForm.h"
using namespace MyFormNamespace;
int main()
{
    using namespace System;
    using namespace System::Windows::Forms;
    Application::EnableVisualStyles();
    Application::SetCompatibleTextRenderingDefault(false);
    Application::Run(gcnew MyForm());
}
```

CylinderCoord.cpp:

```
#include "CylinderCoord.h"
```

```
#include"math.h"
CylinderCoord::CylinderCoord(void)
{
    rad = 0;
    ang = 0;
    z = 0;
}
CylinderCoord::CylinderCoord(double r, double a, double Z) {
    rad = r;
    ang = a;
    z = Z;
}
double CylinderCoord::FindDistance(CylinderCoord^ other) {
    double w1, w2, w3;
    w1 = pow((X() - other->X()), 2);
    w2 = pow((Y() - other->Y()), 2);
    w3 = pow((z - other->z), 2);
    return sqrt(w1 + w2 + w3);
}
double CylinderCoord::FindLineDistance(CylinderCoord^ l1, CylinderCoord^ l2) {
    CylinderCoord^ a1 = gcnwCylinderCoord(l1->X() - X(), l1->Y() - Y(), l1->z - z);
    CylinderCoord^ l1l2 = gcnwCylinderCoord(l2->X() - l1->X(), l2->Y() - l1->Y(), l2-
>z - l1->z);
    double vp = a1->ABSVektMult(l1l2);
    return vp / l1->FindDistance(l2);
}

double CylinderCoord::X() {
    return rad*cos(ang);
}
double CylinderCoord::Y() {
    return rad*sin(ang);
}
```

```
double CylinderCoord::ABSVektMult(CylinderCoord^ other) {
    double x = ang*other->z - z*other->ang;
    double y = z*other->rad - rad*other->z;
    double z = rad*other->ang - ang*other->rad;
    return sqrt(pow(x, 2) + pow(y, 2) + pow(z, 2));
}
bool CylinderCoord::operator ==(CylinderCoord^ other) {
    if (rad == other->rad&&ang == other->ang&&z == other->z) return true;
    else return false;
}
CylinderCoord.h:
#pragma once
refclass CylinderCoord
{
    double rad;
    double ang;
    double z;
public:
    CylinderCoord(void);
    CylinderCoord(double r, double a, double Z);
    double FindDistance(CylinderCoord^ other);
    double FindLineDistance(CylinderCoord^ l1, CylinderCoord^ l2);
    bool operator ==(CylinderCoord^ other);
private:
    double X();
    double Y();
    double ABSVektMult(CylinderCoord^ other);
};
```

Для создания визуального проекта необходима разработка многооконного приложения модульной структуры, включающего формы. На рис. 1 отображено окно визуального проекта в C++.

Цилиндрические координаты

Цилиндрические координаты точки:

Радиус: 1

Угол: 2

Апplikата: 3

Ввод

Посчитать:

Расстояние до другой точки

Расстояние до прямой

Площадь треугольника с вершиной в данной точке

Координаты остальных вершин:

Радиус: 4

Угол: 5

Апplikата: 6

Ввод

Радиус: 7

Угол: 8

Апplikата: 9

Ответ: 24,1315993254905

Рисунок 1. Окно визуального проекта

Объектно-ориентированный стиль программирования позволяет использовать преимущества объектно-ориентированного подхода не только на этапах проектирования и конструирования программных систем, но и на этапах их реализации, тестирования и сопровождения.

Задача научить студентов мыслить объектно является методически сложной. В основе объектно-ориентированной методологии лежат понятия классов и объектов. Классы формируют структуру данных и действий и используют эту информацию для строительства объектов.

Создание объектно-ориентированных проектов моделирования математических систем и структур формирует навыки по формализации задачи, выделению абстракций и объектов данной предметной области и отношений между ними, по созданию объектно-ориентированного программного кода и визуального проекта.

СПИСОК ЛИТЕРАТУРЫ

1. Бадд Т. Объектно-ориентированное программирование в действии /Пер с англ. Спб.: Питер, 1997, 464 с.

2. Буч Г. Объектно-ориентированный анализ и проектирование приложений на C++. М.: Бином; СПб.: Невский диалект, 2000, 560 с.

3. Гайнутдинова Т.Ю., Широкова О.А. Особенности профессиональной подготовки по программированию будущего учителя информатики // Программа и тезисы II Международного форума по педагогическому образованию (МФПО-2016). Казань: Казанский университет, С. 231–232.

4. Иванова Г.С. Технология программирования: учебник. М.: КНОРУС, 2011, 336 с.

5. Мейер Бертран. Объектно-ориентированное конструирование программных систем: пер. с англ. М.: Издат.-торг. дом «Рус. ред.», 2005, 1232 с.

6. Широкова О.А. Объектно-ориентированные проекты решения математических задач // Материалы XI Международной науч.-практ. конф. «Объектные системы — 2015». Ростов н/Д: ШИ (ф) ЮРГПУ (НПИ) им. М.И. Платова, 2015, С. 15–23.

7. Gainutdinova T.U., Shirokova O.A. Features of Professional Teachers Training of Informatics in a Programming Course// Сборник IFTE 2016, Volume XII, Pages 1—451 (July 2016). The European Proceedings of Social & Behavioural Sciences EpSBS – Международный Форум по Педагогическому Образованию, Казань, 2016, С. 30–37.

8. Schildt H. Herb Schildt's C++ Programming Cookbook. Osborne: McGraw Hill, 2008.

OBJECT-ORIENTED PROJECTS WITH THE CREATION OF CLASSES FOR MATHEMATICAL OBJECTS

O.A. Shirokova

Kazan Federal University, Kazan

shirokova2602@mail.ru

Abstract

The article discusses the current problems of software development technology based on an object-oriented programming approach

Keywords: *object-oriented programming, classes, objects.*

REFERENCES

1. *Badd T.* Ob`ektno-orientirovannoe programmirovaniye v dejstvii /Per. s angl. Spb.: Piter, 1997, 464 s.
2. *Buch G.* Ob`ektno-orientirovanny`j analiz i proektirovaniye prilozhenij na S++. M.: Binom; SPb.: Nevskij dialekt, 2000, 560 s.
3. *Gajnutdinova T.Yu., Shirokova O.A.* Osobennosti professional`noj podgotovki po programmirovaniyu budushhego uchitelya informatiki // Programma i tezisy` II Mezhdunarodnogo foruma po pedagogicheskomu obrazovaniyu (MFPO-2016). Kazan`: Kazanskij universitet. S. 231–232.
4. *Ivanova G.S.* Texnologiya programmirovaniya: uchebnik. M.: KNORUS, 2011, 336 s.
5. *Mejer Bertran.* Ob`ektno-orientirovannoe konstruirovaniye programmny`x sistem: per. s angl. M.: Izdat.-torg. dom "Rus. red.", 2005, 1232 s.
6. *Shirokova O.A.* Ob`ektno-orientirovanny`e proekty` resheniya matematicheskix zadach // Materialy` XI Mezhdunarodnoj nauch.-prakt. konf. «Ob`ektny`e sistemy` — 2015». Rostov n/D: ShI (f) YuRGPU (NPI) im. M.I. Platova, 2015, S. 15–23.
7. *Gainutdinova T.U., Shirokova O.A.* Features of Professional Teachers Training of Informatics in a Programming Course// Sbornik IFTE 2016 Volume XII, Pages 1—451 (July 2016) The European Proceedings of Social & Behavioural Sciences EpSBS – Mezhdunarodny`j Forum po Pedagogicheskomu Obrazovaniyu, Kazan`, 2016, S. 30–37.

8. *Schildt H.* Herb Schildt's C++ Programming Cookbook. Osborne: McGraw Hill, 2008.

СВЕДЕНИЯ ОБ АВТОРЕ



ШИРОКОВА Ольга Александровна – кандидат физико-математических наук, доцент, Казанский (Приволжский) федеральный университет, г. Казань.

Olga SHIROKOVA – Ph.D. of Physical and Mathematical Sciences, Associate Professor, Kazan (Volga) Federal University, Kazan.

e-mail: shirokova2602@mail.ru

Материал поступил в редакцию 11 сентября 2019 года