

Данный программный комплекс можно использовать при создании интерактивных виртуальных сред для решения различных задач в морской отрасли.

Список использованных источников:

1. Горб С.И. Тенденции развития технического обслуживания судов // Автоматизация судовых технических средств. Вып. 12. – 2007. – С. 26-34.
2. Грибова В.В., Федорищев Л.А. «Виртуальная реальность в образовании: система разработки интернет-проектов» // НИТО, 2012. – С. 116-118.
3. Соснин П.И. Создание и использование онтологий проектов в разработке автоматизированных систем // КИИ 2010, Тверь. – 2010. Том 2. – С. 187-195.
4. Гаркуша Г.Г. Концепция внедрения е-навигации в практику судовождения // Всеукраїнська конференція «Сучасні методи і моделі розвитку морського транспорту та безпека судноплавства» 26 травня 2017 р. АМІ НУ «ОМА» – С. 84-90.

Шокотько Д.Д.

студентка,

Одеський державний університет

імені І.І. Мечникова

ВИКОРИСТАННЯ ІНДЕКСАТОРІВ ПРИ РЕАЛІЗАЦІЇ АЛГОРИТМІВ ЧИСЕЛЬНИХ МЕТОДІВ ЛІНІЙНОЇ АЛГЕБРИ

Сучасна мова програмування C# та середовище розробки Windows-додатків Microsoft Visual Studio є однією з найпотужніших, процвітаючих та затребуваних систем в IT-галузі [1; 2].

В цьому середовищі розробляються найрізноманітніші програми: від невеликих студентських додатків до масштабних Web-порталів та Web-сервісів.

Але у мови програмування C# також є і специфічні незручності, які вона отримала «у спадок» від своїх іменитих попередників C та C++.

Ми зупинимося на одній з них. Проблема полягає в індексації елементів масивів різної форми. У мові програмування C# вона починається з нуля.

Однак, для позначень, які прийняті в класичній математиці матричних обчислень, де нумерація рядків і стовпців починається з одиниці, це породжує деякі незручності, і саме через це у програмістів часто виникають прикрі помилки.

Цю проблему для задач алгебри матриць, на наш погляд, можна частково вирішити за допомогою індексаторів.

За правилом, в C# індексація масивів різної розмірності реалізується з використанням оператора [].

Ми можемо створити свій тип даних – клас `Class_Matrix`, в якому виконаємо перевантаження цього оператора, не вдаючись до «послуг» методу `operator()`, а за допомогою створення індексатора (`indexer`).

Ми скористаємося основним призначенням індексаторів – підтримувати спеціалізовані масиви, на елементи яких накладається одне або декілька додаткових обмежень.

При цьому самі індексатори можна використовувати в рамках стандартного синтаксису мови C#.

Відомо, що індексатори можуть характеризуватися однією або кількома розмірностями. Але в нашому випадку мова йде про двовірний масив, а ще конкретніше – про квадратні матриці.

Для цього, як вже згадувалося вище, ми визначаємо власний клас, одним з членів якого буде безпосередньо квадратний масив – матриця.

Індексатор надасть нам можливість «працювати» з елементами цієї матриці в індексних межах, які будуть починатися з 1 (як це прийнято в класичній алгебрі матриць та векторів).

Крім того, зручність індексатора полягає в тому, що він програмно забезпечує запобігання некоректного доступу до елементів матриці та дозволяє уникати виняткових ситуацій при можливому виході за індексні межі масивів.

Сконструювавши, за допомогою індексатора, новий тип даних – матрицю, ми можемо, потім, приступити до програмної реалізації базових алгоритмів задач і методів класичної алгебри у вигляді, прийнятому у вищій математиці.

Є цілком зрозумілим, що повний збіг індексних меж та позначень елементів матриць в математиці та в її програмної реалізації, дозволить більш якісно виконувати тестування і налагодження розробленого коду.

Крім цього, ми можемо доповнити додатковими методами і членами сам клас матриць `Class_Matrix`, які зроблять матричні обчислення більш зручними та комфортними для інших розробників.

Нами пропонується наступний варіант такого класу, що описує об'єкт матриця з використанням індексатора:

```
public class Class_Matrix
{
    public double[,] M; // Ссылка собственно на матрицу M
    public int Size; // Ранг матрицы M
    public bool flag; // Флаг последней операции обращения к M[i,j]

    public Class_Matrix(int size) // Конструктор матрицы заданного размера
    {
        Size = size; M = new double[Size, Size];
    }
    // Это – индексатор для класса Matrix
    public double this[int index_i, int index_j]
    {
        get // get-аксесор
        {
            if (ok(index_i, index_j))
            { flag = false; return M[index_i - 1, index_j - 1]; }
        }
    }
}
```

```
else { flag = true; return double.NaN; }  
}  
set // set-аксесор  
{  
if (ok(index_i, index_j))  
{ M[index_i - 1, index_j - 1] = value; flag = false; }  
else flag = true;  
}  
}  
// Метод возвращает true, если оба индекса в пределах размерности M  
private bool ok(int index_i, int index_j)  
{  
bool i = (index_i >= 1 && index_i <= Size);  
bool j = (index_j >= 1 && index_j <= Size);  
if (i && j) return true; return false;  
}  
}
```

На завершення перерахуємо методи алгебри матриць, які були розроблені нами з використанням даного класу: Гаусса, Жордана-Гаусса, Крамера, простих ітерацій, Зейделя, ортогоналізації, обчислення визначника, обчислення оберненої матриці і т.д. [3; 4].

Для користування цими програмними компонентами створене відповідне програмно-методичне забезпечення для студентів фізико-математичних спеціальностей ОНУ імені І.І.Мечникова.

Воно складається зі стислого конспекту (файли .pdf) із алгоритмами зазначених чисельних методів та додається 10 лабораторних робіт з тестовими даними (матрицями), які дозволяють студенту перевірити коректність програм, розроблених власноруч під час виконання індивідуальних контрольних завдань.

Список використаних джерел:

1. Карли Ватсон С# : Издательство «Лори», 2006. – 860 с.
2. Эндрю Троелсен Язык программирования С# 2005 и платформа .NET 2.0, 3-е издание.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2007. – 1168 с.
3. Степан Шахно. Чисельні методи лінійної алгебри: Навч. посібник / – Львів: Видавничий центр ЛНУ імені Івана Франка, 2007. – 245 с.
4. Шахно С.М. Практикум з чисельних методів: навч. посібник / Шахно С.М., Дудикевич А.Т., Левицька С.М. – Львів: ЛНУ імені Івана Франка, 2013. – 432 с.