

ЕДИН МЕТОДИЧЕСКИ ПОДХОД ЗА ОБУЧЕНИЕ ПО ИНФОРМАЦИОННО МОДЕЛИРАНЕ

Никола Вълчанов, Антон Илиев, Тодорка Терзиева

РЕЗЮМЕ

Едно от най-широките приложения на компютърната информатика е изграждането и изследването на информационни и математически модели чрез компютърна симулация. В четвъртата година от обучението си, студентите от специалност „Информатика“ във Факултета по математика и информатика при Пловдивски университет „Паисий Хилендарски“ изучават дисциплината „Информационно моделиране“. Целта на обучението е създаване на теоретични знания, въвеждане на система от понятия и методи за изследване на информационното моделиране, както и изграждане на умения за проектиране и изграждане на среди за симулация на математически модели. В тази статия предлагаме един методически подход за обучение по информационно моделиране.

Ключови думи: динамични системи, математическо моделиране, симулационно моделиране, информационно моделиране.

В съвременното общество за образованието на всеки човек от съществено значение е научната информация за света, която предоставя основни знания за природните, хуманитарните и обществените науки. Изключително важни са уменията и навиците за търсене, извличане, обработка и използване на информация от разнообразен характер, които да се придобиват в процеса на образование и обучение. Една от силните страни на информатиката е нейният интегративен характер, където чрез системен подход могат да се изследват и изучават обекти от различни предметни области.

Една от целите на курса по „Информационно моделиране“ на студенти от специалност Информатика е изграждане на умения за формализация, моделиране и структуриране на информация.

ОСНОВНИ ЕТАПИ ПРИ ОБУЧЕНИЕТО

Обучението по „Информационно моделиране“ преминава през четири основни етапа, които са тясно свързани:

- обзор на системи за симулация - запознаване със съвременни системи за симулация на информационни и математически модели;
- проектиране и разработка на математическа библиотека за симулация, обзор на числени методи на Ойлер и Рунге-Кута за числено решаване на обикновени диференциални уравнения;
- проектиране и разработка на дизайнер на модели – среда за визуално изграждане на математически модели;
- изготвяне на самостоятелен проект – система за симулация на математически модели.

Всеки етап се състои от съвкупност от базови задачи и се базира на знания, получени в предишните етапи.

След всяко занятие се оставят нерешени проблеми за самостоятелна работа. Те могат да бъдат разделени на две категории – задачи за упражнение (повторение на придобитите знания) и задачи за затвърдяване, разширяване и усъвършенстване на усвоените знания, свързани с допълнителни проучвания.

ОБЗОР НА СИСТЕМИТЕ ЗА СИМУЛАЦИЯ

При първоначалният етап на обучение се разглеждат актуални системи за симулация на информационни модели:

- Стелла (Stella) – предоставя набор от графични компоненти за изграждане на Склад (Stock) и Поток (Flow) диаграми, чрез които се представя модела. Уравненията на модела се генерират автоматично от изградената диаграма. Системата има възможност за избор на числен метод, който да бъде използван по време на симулацията на модела (SSS, 2010);
- Венсим (Vensim) – подобна на Стелла. Визуалният редактор предоставя инструменти за параметризация на модела. След приключване на симулацията получените данни могат да бъдат представени на потребителя под формата на графики и таблици. В допълнение Венсим предоставя и инструменти за сравнение на графики, улесняващи търсенето на зависимости, инструменти за сравняване на различни симулации, инструменти за статистическа обработка и др. (VSS, 2010);
- Симплекс (Simplex) – дефиниран е мета език, чрез който се описват моделите. Освен поддръжка на уравнения за описание на математически модели, езикът поддържа дефиниране на събития, мобилни компоненти, йерархични структури, както и поддръжка на масиви, процедури и C интерфейс за имплементация на допълнителна функционалност, като извличане на данни от СУБД, интеграция с външни процеси, работа с файлове, инструменти за математически анализи и др. (SIMPL, 2010);
- Енилоджик (AnyLogic) – предоставя допълнителен набор от визуални инструменти за изграждане на агентно-базирани модели. Резултатите от симулацията могат да бъдат представени таблично, графично или чрез анимация, като системата предоставя компоненти за изграждане на

специфични за модела анимации. Системата включва и набор от вградени функции и възможност да дописване на функционален код в модела (ASS, 2010).

Чрез примерни модели се демонстрират възможностите на всяка една от системите, коментират се предимства и недостатъци. На този етап студентите се сблъскват с реални системи за симулация, чрез които се изграждат, симулират и изследват реални модели. По този начин се активизира познавателната дейност на обучаемите, създават се проблемни ситуации и се стимулира самостоятелното използване на знанията за практически задачи.

ПРОЕКТИРАНЕ И РАЗРАБОТКА НА МАТЕМАТИЧЕСКА БИБЛИОТЕКА ЗА СИМУЛАЦИЯ

На вторият етап се разглеждат числени методи на Ойлер и Рунге-Кута. Обсъждат се отделните стъпки на методите. Прави се анализ по какъв начин те могат да бъдат имплементирани като софтуерно приложение.

Следващата стъпка е кратко въведение в интегрираната среда за разработка Microsoft Visual Studio. Създава се тестова библиотека от класове, която се интегрира в тестово конзолно приложение. На базата на начина на работа на Стелла се правят заключения по какъв начин разглежданите числени методи могат да бъдат интегрирани в математическа библиотека за извършване на симулацията въз основа на математическите модели.

Проектира се примерна математическа библиотека, имплементираща симулация на математически модели чрез числен метод на Ойлер за обикновени диференциални уравнения. Стойностите от всяка стъпка могат да се съхранят в подходяща структура. След приключване на симулацията може да се начертае графика на базата на получените резултати.

Анализът на задачата се извършва под формата на дискусия. Разглеждат се различни подходи за решаване на проблема. Акцентира се върху формализацията и разделянето на задачата на подзадачи. Коментират се предимства от преизползване на код и до колко това опростява системите. Формулира се крайната цел – система за симулация на математически модели и възможностите за разширяване на системата с допълнителни математически библиотеки, които симулират модела чрез друг числен метод. Извършват се следните подзадачи:

- имплементация на алгоритъм за решаване на математически изрази;
- имплементация на парсер, който от текстов вход да извлече диференциалните уравнения и началните стойности за всеки от участниците в модела;
- имплементация на графичен интерфейс за въвеждане на модела във вид на уравнения с начални стойности, симулация на модела и извеждане на крайния резултат под формата на графика.

При разглеждане на първата подзадача се дискутират подходи за изграждане на алгоритми за решаване на математически изрази. Насърчава се креативността, обсъждат се различни решения и се търси най-ефективното.

Често студентите не разбират смисъла от изучаването на стандартни алгоритми и проявяват отрицателно отношение към разглеждането им. По време на разработката на проекти от практиката, обучаемите се изправят пред алгоритмични проблеми, стимулира се техният интерес чрез прилагане принципа за динамична нагледност, за да се демонстрира ползата от стандартни алгоритми и приложението им в нови реални ситуации.

При разглеждане на втората подзадача се обсъждат варианти на описание на математическите модели. Разглежда се и поддръжката на структури от данни, съхраняващи колекции в работната рамка Microsoft .NET (MDPS, 2010). На този етап се акцентира върху изграждане на способност за анализиране на проблеми и за определяне на основните действия за решението им. Постепенното повишаване нивото на сложност на алгоритмичните задачи стимулира студентите, сравнително бързо се виждат резултати от положените от тях усилия. Демонстрира се прилагане на придобитите знания и умения на практика и се създава перспектива от овладяването на нови модерни технологии.

При решаване на третата подзадача се разглеждат средствата за изграждане на графични приложения в Microsoft.NET. Използва се кратък набор от задачи, илюстриращи основните възможности на събитийното програмиране, съобразени с нивото на студентите (ANEVA, 2010). В рамките на тази задача се създава графично приложение, в което се вгражда имплементираната библиотека от класове, получават се резултати от симулацията, които се визуализират.

На този етап студентите сравняват разработените от тях системи с готовите системи като Стелла. Създават се тестови модели, изследват се, като се установява в кои случаи се постига баланс на модела. По този начин се постига по-висока ангажираност и се активизира търсенето и изучаването на допълнителни модели за тестване на изградените от обучаемите системи.

ПРОЕКТИРАНЕ И РАЗРАБОТКА НА ДИЗАЙНЕР НА МОДЕЛИ

При третият етап от обучението се представят среди за изграждане на диаграми, като крайната цел е имплементацията на среда за създаване на Склад и Поток диаграми и интеграцията им с библиотеката за симулация на математически модели.

Студентите се запознават с възможностите за работа с 2D графика в Microsoft .NET и по-специално с GDI+. Тестовите задачи имат за цел въвеждане на технологии, които ще бъдат използвани в следващите задачи. По този начин студентите добиват увереност и подхождат по-смело към следващите задачи.

Проектирането на йерархии от бизнес класове е често подценявана дейност от студентите. В голяма част от случаите те първо изграждат търсената функционалност и след това планират как да я разпределят в бизнес обекти. Считаме, че е необходимо процесът на проектиране да се разглежда детайлно. Обсъждат се шаблони за дизайн, които могат да бъдат използвани в конкретния случай за различни допълнителни дейности като обработка на изключения, каталогизиране на грешки и съобщения и др. Предлагат се възможни надграждания на системата и по какъв начин може да се улесни работата на разработчика при нужда от разширяване. В процеса на проектиране се създават условия за формиране на абстрактното мислене на студентите.

След приключване проектирането на йерархията се преминава към реализация на средата за изграждане на диаграми чрез няколко стъпки.

Една от най-важните стъпки е полагане на основите на йерархията. Прави се обзор на обектно-ориентирания модел в Microsoft .NET. Студентите се запознават със синтаксиса и поддръжката на обектно-ориентирания модел в езика C#. Реализацията на изчертаването се базира на вече разгледания приложен програмен интерфейс GDI. В края на първата стъпка студентите са реализирали приложение с възможност за поставяне на примитиви в диаграма, селектиране на примитив и промяна на местоположението му.

Втората стъпка от имплементацията е реализация на редакция на примитиви. За тази цел се разглеждат два подхода. Единият е имплементация на отделни форми за редакция на различните примитиви. Това е стандартен подход, който не изисква специфични познания, за да бъде реализиран. Другият е динамично изграждане на графичен потребителски интерфейс за редакция на различните примитиви. За осъществяване на този подход се разглежда технологията „отражение“ (reflection), предоставена от работната рамка Microsoft .NET (MDPS, 2010).

Първите две стъпки целят да запознаят студентите с технологиите, които модерните езици за програмиране предоставят. Демонстрират се възможностите на съответния език при проектиране на системата. При това обучаемите:

- придобиват знания и умения за проектиране на йерархии от бизнес обекти;
- прилагат и упражняват разгледаните до момента технологии.

Третата стъпка от имплементацията е реализацията на връзката между примитивите и свойствата за генериране на данни за описания модел. Генерираните данни трябва да са във формат, четим от математическата библиотека. За целта се проектира рекурсивен алгоритъм за обхождане и генериране на информация за примитивите. В края на този етап студентите разполагат със среда за изграждане на диаграми, с възможност за конфигуриране, свързване на примитивите и генериране на четима от математическата библиотека репрезентация на изградения математически модел. Обучаемите изграждат умения за разработване на нестандартни

алгоритми. Подчертава се необходимостта от алгоритмична подготовка при разработка на софтуерни системи.

ПРЕДСТАВЯНЕ НА САМОСТОЯТЕЛЕН ПРОЕКТ

На този етап от обучението по информационно моделиране, студентите вече разполагат с нужните знания за изграждане на среда за симулация на математически модели. Изградени са умения за разработка на приложения чрез работната рамка Microsoft .NET. Интеграцията на математическата библиотека със средата за изграждане на математически модели позволява изграждането на самостоятелния проект, който се изисква в края на курса. Изготвянето му е свързано с надграждане на получената симулационна среда. Всеки студент получава персонално задание, най-често свързано с имплементация на конкретен инструмент за контрол на симулацията, визуализация на данни и др.

В края на курса обучаемите подготвят презентации и демонстрират проектите си пред своите колеги. Създават се условия за свободна дискусия, коментар на срещаните трудности при разработка на проекта, както и обсъждане ефективността на взетите решения.

Провеждането на теоретичната част от изпита се извършва чрез тестова система (PAVLOV, 2010), която автоматично оценява работата на студентите. Използват се лингвистични инструменти (DIAS, 2010).

ЗАКЛЮЧЕНИЕ

Ангажирането на вниманието и активизирането на студентите става все по-трудна задача. При изграждане на системи за симулация чрез математически модели се демонстрират предимствата и възможностите при моделирането на реални обекти и системи. Стимулира се интересът на обучаемите към изучаваните методи и технологии на изследване чрез практическото приложение на натрупаните теоретични знания за решаване на проблеми от различни предметни области.

Чрез средствата и методите на информационното моделиране се развиват умения за:

- анализиране и формализиране на проблеми;
- формулиране на целта на моделирането и определяне на съществените свойства на обекта(процеса), съобразно целта;
- използване на система от понятия и величини, чрез които качествено и количествено се описват свойствата на модела;
- разработване на математически апарат на модела с помощта на съвременни технологии;
- извършване на теоретичен анализ на модела (чрез тестване в различни, теоретично възможни ситуации и изследване на поведението му);

- съставяне на извод за качествата на модела (граница на приложимост, условия за валидност, функционалност и др.) (NIKOLOV, 1993);
- прилагане на модела при решаване на практически задачи и изследване на нови явления.

Обучението, базирано на проекти стимулира творчеството и креативността на студентите и създава условия за възникване на проблемни ситуации. Постепенното увеличаване нивото на сложност на разглежданите задачи спомага за елиминиране на стреса от новата технология. Този подход стимулира активността на студентите и ги предразполага към самостоятелно прилагане на придобитите знания и умения за експериментиране в различни ситуации.

ЛИТЕРАТУРА

ANEVA, S. (2010) A system of fundamental problems for learning event driven programming with the Microsoft Visual C# IDE in the high school. *Proceedings of national conference „Education and the information society”*. Plovdiv: Association „Development of the information society”, pp. 239-251 (in Bulgarian).

NIKOLOV, S. & MAVROVA, R. (1993) *Methods of scientific knowledge*, Makros 2000, Plovdiv (in Bulgarian).

PAVLOV, N. & RAHNEV, A., RAHNEVA, O. (2010) RIA-based system for electronic tests, Scientific conference with international participation "Challenges for higher education and research in a crisis", Sofia, 25-26 June.

DIAS, G. & MORALIYSKI, R. & CORDEIRO, J. & DOUCET A. & AHONEN-MYKA, H. (2010) Automatic Discovery of Word Semantic Relations using Paraphrase Alignment and Distributional Lexical Semantics Analysis, *Journal of Natural Language Engineering, Special Issue on Distributional Lexical Semantics*, R. Basisli and M. Pennacchiotti Eds., Cambridge University Press, ISSN 1351–3249.

ANYLOGIC SIMULATION SOFTWARE (ASS, 2010), <http://www.xjtek.com/> (last visited 20.06.2010).

INTRODUCTION to SIMULATION - SIMPLEX3 (SIMPL, 2010), <http://isgwww.cs.uni-magdeburg.de/sim/its/ws0506/download.html> (last visited 20.06.2010).

MICROSOFT CORPORATION, Provider Model Design Pattern and Specification, (MDPS, 2010), <http://msdn.microsoft.com/en-us/library/ms972319.aspx> (last visited 20.06.2010).

STELLA SIMULATION SOFTWARE (SSS, 2010), <http://www.iseesystems.com/software/Education/StellaSoftware.aspx> (last visited 20.06.2010).

VENSIM SIMULATION SOFTWARE (VSS, 2010), <http://www.vensim.com/software.html> (last visited 20.06.2010).

ONE METHODOLOGICAL APPROACH FOR TEACHING IN INFORMATION MODELING

Nikola Valchanov, Anton Iliev, Todorka Terzieva

ABSTRACT

One of the most extensive applications of computer informatics is the representation and analysis of information and mathematical models using computer simulation. In the fourth year of their studies the students with major in Informatics in Faculty of Mathematics and Informatics at Paisii Hilendarski University of Plovdiv study the discipline „Information modeling”. The purposes of course are creation of the theoretical specificities of the information modeling. It teaches them the skills needed to design and build simulation environments for mathematical models. This paper offers a methodological approach for creating skills for simulation environment development in the students.

Keywords: dynamic systems, mathematical modeling, simulation modeling, information modeling.

Nikola Valchanov
FMI, Plovdiv University
236, Bulgaria Blvd.,
4003 Plovdiv, BULGARIA
e-mail: nvalchanov@gmail.com
&
IMI, Bulgarian Academy of Sciences
1113 Sofia, Acad. G. Bonchev Str., Bl. 8,
BULGARIA

Todorka Terzieva
FMI, Plovdiv University
236, Bulgaria Blvd.,
4003 Plovdiv, BULGARIA
e-mail: dora@uni-plovdiv.bg

Anton Iliev
FMI, Plovdiv University
236, Bulgaria Blvd.,
4003 Plovdiv, BULGARIA
e-mail: aii@uni-plovdiv.bg
&
IMI, Bulgarian Academy of Sciences
1113 Sofia, Acad. G. Bonchev Str., Bl. 8,
BULGARIA