

Список використаних джерел:

1. Ануфриев И.Е., Смирнов А.Б., Смирнова Е.Н. MATLAB 7. – СПб.: БХВ-Петербург, 2005. – 1104 с.: ил.
2. Богданов І.Т. Аналіз резонансних кіл однофазного змінного струму засобами MatLab / І.Т. Богданов, Ю.О. Єфіменко, М.О. Стеценко // Фізика та астрономія в школі. – 2010. – №3/(78).
3. Вейвлет-анализ: основы теории и примеры применения // Успехи физических наук, 1996, Том 166. №11. с.1145-1170.
4. Дремин И.М., Иванов О.В., Нечитайло В.А. Вейвлеты и их использование // Успехи физических наук. – 2001. – №5. – Том 171.
5. Дьяконов В., Абраменкова И. MATLAB. Обработка сигналов и изображений. Специальный справочник. – СПб.: Питер, 2002. – 608 с.
6. Лазарев Ю. Моделирование процессов и систем в MATLAB. Учебный курс. – СПб.: Питер; К.: Издательская группа BHV, 2005. – 512 с.: ил.
7. Лурье М.С., Лурье О.М. Применение программы MATLAB при изучении курса электротехники. Для студентов всех специальностей и форм обучения. – Красноярск: СибГТУ, 2006. – 208 с.
8. Смоленцев Н.К. Основы теории вейвлетов. Вейвлеты в Matlab. – М.: ДМК Пресс, 2005. – 304 с., ил.
9. Скоростецкий О. М. Моделирование та обробка складних сигналів в середовищі MATLAB / Скоростецкий Александр Миколайвич. [Електронний ресурс]. – Режим доступу: <http://s-journal.cdu.edu.ua/base/2008/v2/v2pp254-255.pdf>.
10. Электротехника. Имитационное моделирование в лабораторном практикуме, курсовом и дипломном проектировании: Учебное пособие для студентов. В 2-х частях / М.С.Лурье, О.М.Лурье. – Красноярск: СибГТУ, 2005.
11. Cooper J. A MATLAB Companion for Multivariable Calculus. – Academic Press, 2001.
12. Hunt B., Lipsman R., Rosenberg J., Coombes K., Osborn J., Stuck G.. A Guide to MATLAB. – Cambridge, 2001

In article are considered the advantages and defect of use an wavelet in software complex Matlab Wavelet Toolbox at analysis of spectrums of signals in laboratory practical work on electro and radiotechnican discipline.

Key words: laboratory practicum, physical and technical knowledge, design, Matlab Wavelet Toolbox.

Отримано: 11.07.2011

УДК 378.147:004.94

О. І. Теплицький

Криворізький металургійний факультет Національної металургійної академії України

ЗАСОБИ НАВЧАННЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО МОДЕЛЮВАННЯ СТУДЕНТІВ ПРИРОДНИЧИХ СПЕЦІАЛЬНОСТЕЙ ПЕДАГОГІЧНИХ УНІВЕРСИТЕТІВ

У статті розглядаються соціально-конструктивістські та конструкціоністські основи об'єктно-орієнтованого моделювання. Обговорюється взаємодія учасників навчального процесу у соціально-конструктивістському середовищі. Конкретизуються засоби навчання об'єктно-орієнтованого моделювання.

Ключові слова: об'єктно-орієнтоване моделювання, конструктивізм, конструкціонізм, Alice, Web 2.0.

Постановка проблеми. Розвиток технологій програмування зумовлює необхідність практичного вивчення не лише сучасних програмних засобів, але й технологій їх розробки. Проте разом із зростанням можливостей середовищ розробки зростає їхня складність, і як наслідок – зростає складність їх вивчення, що може призводити до появи технологічного ухилу в навчанні програмування. Одним із шляхів вирішення проблеми фундаменталізації навчання програмування майбутніх учителів природничих дисциплін є введення в курс інформатики *об'єктно-орієнтованого моделювання*. В процесі навчання програмування майбутніх учителів володіння засобами об'єктно-орієнтованого моделювання допомагає об'єднати технології об'єктно-орієнтованого, подієорієнтованого та візуального програмування в єдиному середовищі [10].

Аналіз останніх досліджень з вирішення загальної проблеми та виділення невирішених питань. Питанню навчання моделювання майбутніх учителів присвячені роботи багатьох дослідників. Зокрема, у роботах О. Г. Гейна, Н. В. Макарової, О. П. Шестакова розкриваються методичні аспекти навчання інформаційного моделювання, в роботах О. П. Одинцової, Т. В. Софронової, О. А. Тарасової – геометро-графічного моделювання; в роботах А. В. Бобровської, Л. В. Кавурко, І. Г. Обойщикова – математичного моделювання; в роботах Д. Д. Бичкової, П. В. Кійка, І. В. Паболкова, І. О. Теплицького, Л. М. Шенгелії показана інтегративна роль комп'ютерного моделювання.

Методичні засади застосування об'єктно-орієнтованого підходу в процесі підготовки майбутніх вчителів розглянуті в роботах І. М. Аржанова, Є. В. Баранової, Ю. С. Рамського, О. Г. Степанова. Незважаючи на відсутність єдиного підходу до визначання об'єктно-орієнтованого моделювання, питання його навчання об'єктно-орієнтованого моделювання розглядаються в роботах зарубіжних дослідників Й. Бйорстлера, Т. Брінди, В. Неллеса, Е. Дж. Корнецькі, Ж.-П. Ріго, С. Хад'єруїта; зокрема, у дослідженні [1] представлена дидактична система навчання об'єктно-орієнтованого моделювання в середній школі. Проте цілісна методична система навчання об'єктно-орієнтованого моделювання майбутніх учителів природничо-математичних дисциплін є не розробленою.

Опрацювання теоретичних аспектів педагогіки Заходу показує, що найбільш прогресивні з них спираються на основні положення теорії соціального конструктивізму, філософські, соціологічні та педагогічні аспекти якої розглядалися у роботах Е. фон Глазерсфельда, К. Дж. Джерджена, П. Лоренцена, А. В. Пашкової, К. В. Рибачука, С. А. Цоколова, М. А. Чошанова. Програмні засоби технології соціального конструктивізму розглядалися в роботах Дж. Адамса, Т. М. Брусенцової, А. Гольдберг, М. Гуздіала, В. Данн, Дж. В. Максвелла.

Метою статті є аналіз програмних засобів навчання об'єктно-орієнтованого моделювання студентів природничих спеціальностей педагогічних університетів.

Виклад основного матеріалу. Конструктивізм та його різновид – конструкціонізм – помітно вплинув на педагогічний дизайн і втілюється в наступних педагогічних середовищах: Logo і його похідні NetLogo та StarLogo; Squeak і похідний від нього Scratch; Alice та ін.

Назва Logo (LOGO) походить від грецького «логос», що означає «слово», «сенс», «ідея». У літературі термін «Logo» використовується в двох значеннях: 1) як мова програмування, настільки проста, що її можуть освоїти діти, але настільки потужна й виразна, що й досвідчений програміст знайде в ній багато цікавого; 2) як філософія навчання, система поглядів на процес навчання, покликана, як вважає С. Пейперт, не частково поліпшити, а корінним чином революціонізувати традиційну організацію навчання. Мова Logo займає в цій системі центральне місце.

Філософія навчання Logo передбачає перетворення традиційної системи навчання і ґрунтується на ідеї «використання комп'ютера як моделі, яка може вплинути на наш спосіб мислення про самих себе» [9, с.26]. Об'єкти середовища Logo – це комп'ютер, що «математично говорить за істоту» [9, с.47] і Черепаха, «кібернетична тварина», керована за допомогою комп'ютера. Навчання в Logo відбувається в процесі «бесід» учня з Черепахою і комп'ютером; на відміну від традиційної організації навчання, в Logo не комп'ютер управляє процесом навчання, а учень «навчає» комп'ютер, «говорячи йому» мовою Logo. Черепаха, як і комп'ютер, – об'єкт в середовищі Logo. Надсилаючи ко-

манди-повідомлення об'єктові «Черепаха», учні у природний спосіб засвоюють принципи об'єктно-орієнтованого програмування. «Робота в мікросвіті Черепахи – це модель вивчення ідей у той самий спосіб, у який ми пізнаємо іншу людину. Учні, котрі працюють в цьому середовищі, безумовно, відкривають в ньому цікаві факти, приходять до узагальнень, засвоюють навички» [9, с.136].

На початку 1990-х років М. Резнік запропонував використати мультиагентне співтовариство черепашок для освоєння учнями екологічних стратегій [5]. З безліччю черепашок у мові StarLogo учні могли спостерігати, вивчати й моделювати складні фізичні, хімічні, біологічні й соціальні феномени. Хоча мова створювалася в першу чергу як засіб навчання, у цьому середовищі виявилось можливим ставити й серйозні експерименти з мультиагентного моделювання. Дослідницькі можливості StarLogo одержали подальший розвиток у NetLogo, в якому у останні роки були побудовані різні дослідницькі моделі, які використовувалися в наукових статтях і обговорювалися в книгах з мультиагентного моделювання й соціології [6].

Бібліотека моделей, створених у середовищі NetLogo, велика й поповнюється не тільки розроблювачами, але й членами співтовариства (логосфери), у якому можна: прочитати опис моделі, її призначення, покладені в основу принципи; подивитися виконання програми в мережі; скачати модель і запустити на своєму комп'ютері; внести до моделі зміни й використати готові процедури, узяті з чужої моделі, для своїх власних потреб; завантажити свою модель на сервер і запропонувати її до обговорення й спільного використання.

У 1968 році А. Кей, надиханий ідеями С. Пейперта, придумав Dynabook – «персональний комп'ютер для дітей будь-якого віку» [2]. А. Кей бачив роль персонального комп'ютера як особистісного динамічного середовища (метамедіа), що об'єднувало в собі всі інші середовища: текст, графіку, анімацію і навіть те, що ще не винайдено [3]. Так само, як і Лого, мова Smalltalk, розроблена як програмна частина проекту Dynabook, є одночасно і мовою програмування, і середовищем розробки програм. Це чисто об'єктно-орієнтована мова, у якій абсолютно все розглядається як об'єкти. Як зазначає один з її розробників Д. Інгаллс, «мета проекту Smalltalk – зробити світ інформації доступним для дітей будь-якого віку. Всі труднощі полягають у тому, щоб знайти й застосувати досить прості й ефективні метафори, які дозволять людині вільно оперувати найрізноманітнішою інформацією від чисел і текстів до звукових і зорових образів» [4]. В основу мови покладені дві прості ідеї: 1) усе є об'єктами; 2) об'єкти взаємодіють, обмінюючись повідомленнями. Сучасною реалізацією Smalltalk є Squeak [11], в якому з'являється усе більше властивостей проекту Dynabook – потужна 2D- і 3D-графіка, багатоголосний і синтезований звук, підтримка анімації й відео, засоби для роботи з різноманітними медіа-форматами тощо [8].

Scratch – це одне середовище програмування, створене під керівництвом М. Резніка [7]. Scratch дозволяє дітям створювати власні анімовані й інтерактивні історії, ігри й інші творіння. Основне завдання проекту – стати часткою освітньої програми для дітей і підлітків, розвинути в них творчі здібності, логічне мислення і свободу у використанні інформаційних технологій. Все це пропонується розвинути шляхом залучення учнів до процесу конструювання інтерактивних презентацій, мультимедіа, ігор. Діти можуть складати свої програми з блоків команд («цеглинок») так само, як вони будували будиночки і машинки з деталей «Лего». За своєю внутрішньою архітектурою Scratch базується на Squeak, тому при «вичерпанні» можливостей Scratch у міру розвитку навичок програмування можна перейти до батьківського мультимедійного середовища об'єктно-орієнтованого моделювання Squeak, використовуючи потужні засоби ООП мови Smalltalk. Так само, як і у NetLogo, навколо Scratch існує співтовариство, схема взаємодії учасників якого представлена на рис. 1.

Scratch привчає збирати проект із цеглинок і ділитися результатами своїх дій з іншими людьми. Ці навички важливі не тільки усередині спеціальних середовищ програмування,

але й у сучасних мережних співтовариствах. Єдність процесів створення, пошуку й зберігання інформаційних цеглинок всі частіше можна спостерігати на сторінках сучасних сайтів, що використовують концепцію Web 2.0. Метафора будівельних блоків, з яких діти й дорослі можуть збирати прості й дуже складні конструкції, є присутнім не тільки в навчальних проєктах, але й у більшості сучасних мережних сервісів форматів Web 2.0, призначених для підтримки організацій і мережних співтовариств обміну знаннями. М. Резнік, описуючи педагогічні можливості мови Scratch, використовує метафору спіралі творчого розвитку, представлену на рис. 2 [7]: 1) люди уявляють і представляють, що саме вони хочуть зробити й одержати в результаті; 2) люди створюють проєкт, заснований на своїх поданнях; 3) люди грають із результатами своєї діяльності; 4) люди діляться результатами своєї діяльності з іншими людьми; 5) люди обмірковують і обговорюють свої результати; 6) обговорення й обмірковування приведе до нових подань і нових проєктів.

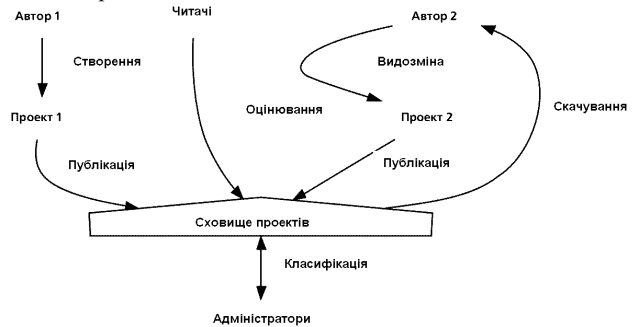


Рис. 1. Схема взаємодії в логосфері NetLogo та Scratch

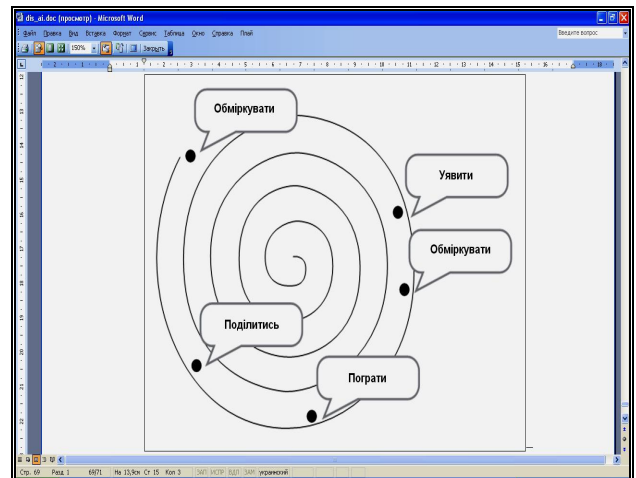


Рис. 2. Спіраль творчого розвитку

Середовище розробки Alice, так само, як і Scratch – відносно новий проєкт, що розробляється в університеті Карнегі-Меллона. На відміну від своїх попередників, Alice – повністю тривимірне середовище моделювання. Alice 2.2 позиціонується розробниками як засіб навчання об'єктно-орієнтованого програмування, а Alice 3, що станом на вересень 2011 р. знаходиться на стадії бета-тестування – як засіб об'єктно-орієнтованого моделювання [10].

Alice є середовищем, в якому можна маніпулювати 3D-об'єктами (рухати, обертати, міняти колір і т.д.) і створювати програми, що генерують анімацію у віртуальних світах. Робоча площина Alice розбита на декілька вікон: у одному в реальному часі відображується віртуальний світ; у іншому присутнє дерево об'єктів, і для кожного з них надається набір доступних методів, функцій та властивостей; центральна частина відведена під редактор вихідного коду.

Запропонована в Alice концепція навчання фактично є зануренням у світ об'єктно-орієнтованого моделювання без яких-небудь істотних і часто непотрібних спрощень цієї парадигми – маніпулювати об'єктами можна тільки за допомогою їх властивостей, функцій та методів, як убудованих, так і сконструйованих користувачем. В Alice максимально скорочений обсяг введення з клавіатури: для пере-

важної більшості дій досить миші. Код програми не є текстом в звичному розумінні: в межах одного методу він є набором вкладених блоків, виділених кольором залежно від типу (цикли, умовні переходи й ін.), їх можна згорнути, перетягувати, змінюючи порядок, і т.д. Незважаючи на таке полегшення, мовою програмування в Alice є не навчальна, а професійна мова Java. Проте середовище розробки Alice надає можливість відображувати створену програму як у стилі мови Java, так і в стилі мови Smalltalk.

Висновки:

1. Фундаменталізація інформатичної освіти вимагає перебудови процесу навчання на основі широкого застосування фундаментальних концепцій інформатики: моделювання, теорії систем та об'єктно-орієнтованого підходу, що разом утворюють якісно нову концепцію – об'єктно-орієнтоване моделювання.

2. Методологічною основою побудови методичних систем навчання об'єктно-орієнтованого моделювання є педагогічна філософія соціального конструктивізму. Грунтуючись на засадах вітчизняної педагогічної психології, вона втілює в собі демократичний підхід до освіти, особистісну зорієнтованість, компетентісний прагматизм, розвиток дивергентного критичного мислення, навчання у спільноті та через спільноту.

3. Соціально-конструктивістські засоби навчання об'єктно-орієнтованого включають три групи програмних засобів: соціально-конструктивістські середовища об'єктно-орієнтованого моделювання, соціально-конструктивістські системи підтримки навчання та соціально-конструктивістські засоби Web 2.0.

Перспективи подальших досліджень: розробка методичних основ навчання інформатичних дисциплін на основі технологій соціального конструктивізму.

Список використаних джерел:

1. Brinda T. Didaktisches System für objektorientiertes Modellieren im Informatikunterricht der Sekundarstufe II : Doktor der Naturwissenschaften (Dr. rer. nat.) genehmigte Dissertation / Torsten Brinda ; Universität Siegen. – Siegen, 2004. – 279 p.
2. Kay A. A Personal Computer for Children of All Ages [Electronic resource] / Alan C. Kay // Proceedings of the ACM National Conference. – Boston, 1972. – Mode of access : <http://www.history-computer.com/Library/Kay72.pdf>.

УДК 378.147:004.94

І. О. Теплицький, С. О. Семеріков

Криворізький національний університет

МОДЕЛЮВАННЯ ЗА ДОПОМОГОЮ ВИПАДКОВИХ ЧИСЕЛ

Статтю присвячено методиці побудови та дослідження стохастичних моделей на основі методу Монте-Карло. Розглядається модель броунівського руху, побудова й опрацювання якої вводить у світ випадкових чисел і математичної статистики, сприяє формуванню уявлень про розподіли ймовірностей, зокрема ілюструє два поширених розподіли: рівномірний та нормальний.

Ключові слова: комп'ютерне моделювання, метод Монте-Карло, випадкові числа, рівномірний розподіл, електронні таблиці.

Постановка проблеми. Зазвичай перебіг багатьох процесів визначається строгими й чіткими закономірностями: значення вихідних параметрів однозначно залежать від значень відповідних величин на вході (їх початкових значень). Ці закономірності подаються математичним записом у вигляді точних формул. Явища, що описуються такими величинами, мають назву детермінованих (від латинського *determino* – визначати, обумовлювати), таку ж назву мають і відповідні моделі. Проте, окрім детермінованих процесів і явищ, існують і такі, що для них неможливо за допомогою точних формул врахувати різноманітні впливи випадкових факторів. Їхні характеристики за своєю природою можуть набувати лише випадкових значень. Такі величини називають випадковими або *стохастичними* (від грецького *stochasticos* – той, що вміє вгадувати, випадковий). Цю ж назву – «стохастичні» – мають і математичні моделі, що містять такі величини. Якщо в детермінованих явищах багаторазово відтворювати ті самі початкові умови, то обов'язково від-

3. Kay A. Personal Dynamic Media / Alan Kay and Adele Goldberg // Computer. – 1977. – Vol. 10, Issue 3. – March. – P. 31–41.
4. Maxwell J. W. Tracing the Dynabook: A Study of Technocultural Transformations : PhD Dissertation / John W. Maxwell ; The University of British Columbia. – Vancouver, 2006. – VIII+303 p.
5. Resnick M. Thinking Like a Tree (and Other Forms of Ecological Thinking) / Mitchel Resnick. – International Journal of Computers for Mathematical Learning. – 2003. – Vol. 8, No. 1. – P. 43–62.
6. Resnick M. Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel Microworlds / Mitchel Resnick. – Cambridge : The MIT Press, 1997. – 181 p.
7. Scratch: Programming for All / Mitchel Resnick, John Maloney, Andres Monroy Hernandez, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, Yasmin Kafai // Communications of the ACM. – 2009. – Vol. 52, No. 11. – P. 60–67.
8. Лесневский А. С. Объектно-ориентированное программирование для начинающих / А. С. Лесневский. – М. : БИНОМ. Лаборатория базовых знаний, 2005. – 232 с.
9. Пейперт С. Переворот в сознании: дети, компьютеры и плодотворные идеи / Сеймур Пейперт. – М. : Педагогика, 1989. – 224 с.
10. Теплицький О. І. Об'єктно-орієнтоване моделювання в Alice. Частина 1 / О. І. Теплицький ; за наук. ред. акад. НАПН України М. І. Жалдака. – К. : НПУ імені М. П. Драгоманова, 2011. – 56 с.
11. Теплицький О. І. Динамічне графічне об'єктно-орієнтоване моделювання в мультимедіа-середовищі мобільного навчання Squeak / О. І. Теплицький, І. О. Теплицький, С. О. Семеріков // Науковий часопис Національного педагогічного університету імені М.П. Драгоманова. Серія №2. Комп'ютерно-орієнтовані системи навчання : зб. наук. пр. / Редація. – К. : НПУ імені М.П. Драгоманова. – №7 (14). – 2009. – С. 49–54.

The article deals with social-constructivist and constructionist fundamentals of object-oriented modeling. We discuss the interaction of participants of educational process in the social-constructivist environment. The learning tools of object-oriented modeling are specified.

Key words: object-oriented modeling, constructivism, constructionism, Alice, Web 2.0.

Отримано: 28.05.2011

творюватимуться ті самі результати. У випадку стохастичних процесів результати кожного разу будуть новими.

Аналіз останніх досліджень з вирішення загальної проблеми та виділення невирішених питань. У роботах [2–7] розглянуті основні елементи педагогічної технології комп'ютерного математичного моделювання, систематично викладеної у навчальному посібнику [8]; наводяться численні приклади її застосування до побудови й дослідження детермінованих навчальних моделей у середовищі електронних таблиць.

Метою статті є розгляд методики побудови стохастичних моделей.

Виклад основного матеріалу

1. Метод Монте-Карло.

Існують різні підходи до моделювання систем, що містять стохастичні характеристики, але найбільш простим і поширеним є метод випадкової вибірки або метод Монте-