

теріалу, за який учень отримав найменшу кількість балів. У цій же підпрограмі відбувається і оцінювання результатів вивчення пройденого розділу. Це відбувається з врахуванням отриманих відповідей на задачі і тести. Якщо все було розв'язано добре або в розділ не було включено засобів діагностики, то виставляється максимальна оцінка 100 балів. У випадку не отримання жодної правильної відповіді учень отримує нуль балів.

1. Clancey W.J. The role of qualitative models in instruction//Self J.(ed.) Artificial intelligence and human learning.-London:Chapman and Hall, 1988.-p.49-68.
2. Крук М.М., Семотюк В.М. Реалізація навчальної системи керуючого типу.// - Вісник ДУ "Львівської політехніки"/ 1998, вип.351, с. 29-33.

© В.Семотюк, О.Талалай
 Національний університет "Львівська політехніка"

УДК 681.3:658.56

ПРИНЦИПИ ПОБУДОВИ ЕКСПЕРТНОЇ НАВЧАЛЬНОЇ СИСТЕМИ З ПРОГРАМУВАННЯ НА НИЗЬКОМУ РІВНІ

© Семотюк В., Талалай О., 2002

Пропонуються підходи, що реалізовані при побудові експертної комп'ютерної навчальної системи для вивчення програмування мовою низького рівня. Розглядаються питання опису вхідних і вихідних параметрів, архітектури експертної навчальної системи, опису даних та їх структури, алгоритми знаходження розв'язку, алгоритм та програма створення зв'язків між поняттями навчального матеріалу.

The approach to the computer expert tutor system design is proposed. The description of data and system parameters, the system architecture, the algorithms of creating relations between items and terms is considered.

У загальному випадку задача керування навчанням зводиться до задачі планування навчальних впливів для досягнення деякої цілі, виходячи з поточного стану знань і вмій учня [1]. Вхідними даними для підсистеми керування є початкова (базова) модель учня, яку методами навчального впливу необхідно наблизити до такої моделі, яка б відповідала кінцевій меті навчання. З цього випливає інший важливий параметр - ме-

та навчання, яка повинна описувати потрібні знання та вміння учня. Саме ці знання та вміння він повинен отримати в процесі роботи із системою. В нашому випадку метою навчання є явно задана середня оцінка успішності учня. Ця оцінка вводиться самим користувачем при реєстрації його в системі і може набувати значення за стобальною системою (0-100). Проте це не означає, що учневі (особі, яка навчається) достатньо набрати середній бал, більший чи такий бал, що введено при реєстрації, і при цьому опустити деякі невивчені теми. Йому все ж потрібно вивчити весь навчальний матеріал і наблизити свою модель до моделі учня, що відповідає такому середньому балу.

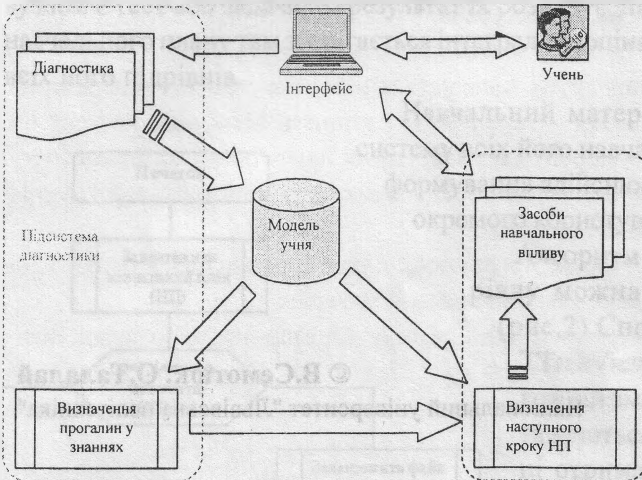


Рис. 1. Структура експертної навчальної системи

Опишемо алгоритм знаходження розв'язку, який повинен бути динамічним процесом, що спрямований на досягнення цілі навчання, виходячи з поточного стану знань і вмінь особи, що навчається. Для досягнення цілі будується план навчання, що складається з окремих кроків. Кожен крок є в загальному випадку послідовністю описаних вище навчальних впливів. Навчальний матеріал, що є в базі знань, а також кількість повторного його вивчення являють собою деякі обмеження на процес навчання. Тобто, навчання може завершитись в зв'язку з тим, що пройдено весь матеріал з бази знань або кількість його повторного вивчення перевищила допустимі норми. Після того, як навчальний план сформовано, він покроково виконується. Якщо в процесі виконання плану виникає ситуація, коли виконання наступного кроку є неможливим або недоцільним, то підсистема не модифікує чи генерує новий план, а лише визначає той його елемент, на який потрібно зробити перехід. На рис.1 наведено загальний вигляд структури експертної навчальної системи, де виділено окремо блоки, що входять до підсистеми керування навчанням.

У процесі роботи система використовує динамічні структури для збереження та оперування даними, а для збереження в постійній пам'яті – структуровані файли. При роботі з цими файлами застосовуються принципи реляційних баз даних. Така організація збереження даних приводить до економії пам'яті на магнітних носіях інформації. Досягається це за рахунок того, що масиви даних, що зустрічаються, дуже часто збері-

Для досягнення поставленої мети програма керування повинна вміти оперувати набором операцій - навчальних впливів, якими є:

- виведення навчальної інформації;
- уточнюючі дані;
- виведення умов тестів і задач;
- введення результатів і відповідей.

Набір цих операцій дозволяє керувати пізнавальною діяльністю учня і змінювати його поточну модель, отже, веде його до поставленої мети.

гаються в окремих таблицях. Звертання до потрібної інформації здійснюється шляхом приєднання таблиці через спеціальні поля доступу. Процес приєднання показано на рис.2.

Отже, повернемося до даних, якими оперує програма: навчальний план і зв'язки між поняттями і знаннями бази. Навчальний план це – послідовність кроків, які необхідно пройти для набуття знань з даної предметної області. Інформація зберігається в таблиці PLAN.

Система зв'язків безпосередньо пов'язана з алгоритмом керування навчанням і призначена для опису співвідношень між поняттями і знаннями[2]. В процесі роботи програма буде використовувати їх для переходів між темами. Цю систему можна створювати щодо кожного пункту навчального плану. Тому і формується для кожного пункту окрема таблиця, яка має розширення "rlt".

Нижче наведено дві основні структури для зберігання цих даних в структурованих файлах. Треба зазначити, що під час виконання програми дані зберігаються у вигляді дерева, тому структури побудовані так, щоб зберегти цю ієрархію в таблицях.

Призначення полів:

```

k_plan
• k_own } - описує навчальний план. Таблиця типу PLAN
name[100]

```

- k_plan - первинний ключ таблиці PLAN;
- k_own - поле, що визначає "батьківський" вузол;
- name - назва елемента навчального плану.

```

K_id
k_relate
• k_own } - описує систему зв'язків. Таблиця типу RELATE
K_obj
name[100]

```

- k_id - поле, призначене для ідентифікації об'єкта;
- k_relate - первинний ключ таблиці;
- k_own - адреса елемента, вищого за ієрархією;
- k_obj - первинний ключ елемента, що зберігається в цьому записі;
- name - буфер для зберігання службової інформації.

Як зазначено вище, в процесі виконання система динамічно будує дерева даних, тому контейнером для них був вибраний об'єкт TTreeView, призначений для представлення ієрархічних даних.

Розглянемо модуль створення зв'язків.

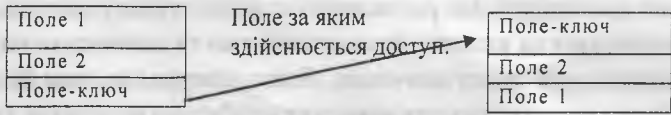


Рис. 2. Схема приєднання таблиці

Щоб правильно проводити навчання, система керування повинна знати, як одне поняття пов'язане з іншим. Такі зв'язки може створити лише експерт, тому йому необхідно надати інструмент для цього. Ним є програмний модуль Relations. Він призначений насамперед для роз'яснення кожного пункту навчального плану. Тобто з його допомогою можна вказати, якими знаннями та навичками потрібно володіти, щоб вважати відповідний пункт вивченим. Якщо, наприклад, ним буде поняття "Регістри", тоді доцільно вказати, що для вивчення необхідно оволодіти такими знаннями:

- реєстри загального призначення;
- сегментні реєстри;
- реєстри прапорців.

Тепер ми знаємо, які знання містить даний тест. Зазначимо, що той самий тест (задача) в іншому розділі навчального плану може мати вказівку на зовсім інші знання, тобто може залежати від контексту. Маючи такі дані, ми можемо судити про рівень засвоєння тих чи інших знань.

Система може працювати з двома типами баз знань:

- 1) база, що містить знання для учнів-початківців;
- 2) знання для учнів, що мають певні навичками у даній предметній області.

Тому перше, що робить програма – це з'ясує, з якою базою знань експерт буде працювати. Залежно від зробленого вибору завантажувється потрібний навчальний план. Далі, коли експерт вибрав пункт навчального плану, зв'язки якого він хоче редагувати, завантажуються дані з відповідної таблиці. Коли таких даних не існує, таблиця створюється, і програма входить в режим створення зв'язків.

Алгоритми роботи з деревом передбачають операції додавання нового вузла, знищення вже існуючих, зміну порядку розташування і, як результат всієї роботи, запис зроблених змін. Якщо зроблені зміни не були збережені і команда виходу з програми від користувача не надходила, то запис буде зроблений автоматично. Система працює так доти, поки ненадійде команда від користувача про вихід з програми.

Під час виконання якихось дій над деревом, що зображає систему навчальних впливів для певного кроку навчального плану, програма модифікує відповідні дані в таблиці зв'язків. Такі зміни спершу зберігаються в оперативній пам'яті і лише після команди запису вони будуть збережені в таблиці на диску. Команда завантаження файлу зі зв'язками створює віртуальну таблицю в пам'яті і надалі вся робота виконується над нею.

Команди знищення елемента, зміна його розташування є лише візуальними і до зміни віртуальної таблиці не призводять. Ці зміни будуть відображені лише після запису в основні таблиці на диску. Зроблено це через те, що знищення даних, зміна порядку в нашому випадку не потребують контролю.

Основною операцією є додавання нового елемента. Саме вона модифікує таблицю в оперативній пам'яті, створюючи новий запис з відповідними реквізитами. Кожному елементу дерева відповідає запис у віртуальній таблиці. Збереження інформації на диск – це перебір всіх вершин дерева і запис даних з таблиці в пам'яті, які йому відповідають.

2. Семотюк В.М., Овсяк В.К., Крук М.М. Управління базами даних комп'ютерної навчальної системи керуючого типу. - Матеріали 2-ї міжнародної конференції "Інформаційні технології друкарства: Алгоритми, системи, сигнали - Друкотехн98". - Львів, 1998, стор. 32-35.

Н.Павич

Національний університет "Львівська політехніка"

УДК 004.942:534-8

ІМІТАЦІЙНА МОДЕЛЬ ВИЗНАЧЕННЯ ФОРМИ ОБ'ЄМНОЇ НЕОДНОРІДНОСТІ У ПРУЖНОМУ СЕРЕДОВИЩІ

© Павич Н., 2002

Запропоновано структуру імітаційної моделі ультразвукового контролю пружного середовища з об'ємною неоднорідністю, яка комплексно забезпечує моделювання процесу визначення форми поверхні неоднорідності. Розглянуто функції та особливості основних складових структури імітаційної моделі.

The model structure of ultrasonic non-destructive testing of an elastic medium with volumetric inhomogeneity which complex provides determination process modelling of inhomogeneity surface form is offered. Functions and futures of model structure basic components is considered.

Важливою проблемою при виготовленні та експлуатації різноманітних елементів конструкцій і систем, зокрема елементів атомних реакторів, виробів аерокосмічної галузі, хімічної промисловості тощо, є виявлення та ідентифікація прихованих внутрішніх дефектів типу тріщин, порожнин та різного типу неоднорідностей. Для контролю пружних середовищ одними з найефективніших є методи, що використовують ультразвукові хвилі. При цьому виникає задача моделювання процесів ультразвукового контролю, оскільки у порівняно з експериментальними роботами, моделювання вимагає менших затрат на реалізацію. Для багатьох випадків важливо знати характеристики захованих дефектів, особливо об'ємних, визначити їх форму. Складність даної проблеми обумовлена тим, що не існує простих аналітичних залежностей між шуканими характеристиками неоднорідностей та параметрами, що вимірюються безпосередньо.

Особливості процесів ультразвукового контролю пружних середовищ з об'ємними неоднорідностями визначають склад та структуру імітаційної моделі, загальна струк-