

ПРОЕКТУВАННЯ НАВЧАЮЧИХ МОДУЛІВ ПРОГРАМНОГО КОМПЛЕКСУ ПОШУКУ ОПТИМАЛЬНИХ РІШЕНЬ

С.В. Бевз

Україна, м. Вінниця
Вінницький державний технічний університет
Тел.: (043) 35-70-23.

Розглянуто основи побудови програми проектування навчаючих та тестових форм, що слугує основою для функціонування навчаючої та тестової програм програмного комплексу пошуку оптимальних рішень. Подано основні структурні взаємозв'язки між програмами та розглянуто функціональні можливості проектування навчаючих і тестових форм. Проаналізовано структуру банку даних, який охоплює навчаючі і тестові модулі та базу даних інформації про них і взаємозв'язки між ними. Пояснено систему доступу до банку даних та захисту від несанкціонованого доступу до даних тестового контролю знань з боку користувачів.

Вступ.

Для сьогодення характерний повсюдний перехід до побудови інформаційного суспільства, що, звісно, висуває нові вимоги до освітянської галузі. Інформатизація освіти насамперед передбачає впровадження новітніх інформаційних технологій у сфері навчання, удосконалення форми і змісту навчаючого процесу, впровадження комп'ютерних методів вивчення матеріалу та тестування отриманих знань, що визначає широке застосування навчаючих програм, надто для дистанційних форм навчання. Такий підхід дозволяє розв'язувати проблеми освіти на найвищому рівні, з урахуванням висунутих до неї вимог. Проте поряд з перевагами навчаючих програм, як-от: це забезпечення безперервного рівномірного навчаючого потоку інформації, можливість вибору необхідного матеріалу та оптимального темпу навчання, організація системного контролю знань — простежуються й деякі недоліки, а саме: втрачається індивідуалізація процесу навчання, можливість врахування психофізіологічних особливостей кожного слухача, оскільки заздалегідь обраний алгоритм навчаючої програми, як правило, не може бути змінений засобами цієї програми з метою адаптації останньої до можливостей окремого слухача, а набору прикладів та тестових завдань, які теж можуть вимагати ітераційного уточнення, не завжди достатньо для формування цілісного уявлення про предмет дослідження.

Програмний комплекс пошуку оптимальних рішень.

Для усунення вищезгаданих недоліків до програмного комплексу (ПК) пошуку оптимальних рішень (ПОР) поряд з навчаючою, тестовою і розрахунковою програмою включено та програму проектування навчаючих модулів.

Навчаюча програма репрезентує теоретичні основи критеріального методу [1]. Матеріал для вивчення може бути подано в текстовому чи динамічному режимах. В останньому випадку передбачено диференціацію режиму на автоматичний чи діалоговий. У діалоговому режимі перехід між сторінками навчаючих модулів і між самими навчаючими модулями здійснюється як у прямому, так і в зворотному напрямках. Користувач може призупинити динамічний перегляд модулів та перегортати вперед чи назад сторінки навчаючої програми.

Контроль знань здійснюється програмою тестування. Користувачеві пропонується 6 тестових завдань, які посортовані за рівнями складності. Програмою випадковим чином вибирається по 2 завдання з бази даних кожного рівня. Залежно від повноти і правильності, від рівня складності завдання, а також від часу, який затрачено на обдумування тесту, формується відповідна кількість балів. Якщо відповіді до тестів були не правильні або не повні, то пропонується повторна спроба відповіді. Максимальна кількість спроб — три. При цьому з кожною наступною спробою кількість балів за правильну відповідь пропорційно зменшується. Окрім іншого, звісно, витрачається додатковий час на повторні спроби, що також впливає на оцінку відповіді. З огляду на це слухач може відмовитися від повторної спроби відповіді і зосередитися на наступних питаннях.

У програмах розрахункового модулю реалізуються нові та класичні методи пошуку і аналізу оптимального розв'язку, прогнозування, уточнення результату на ґрунті критеріального моделювання. Використання в програмі альтернативних методів розв'язання задач (симплекс-метод, метод дихотомії, метод золотого перерізу, квадратичної інтерполяції, безпосереднього розв'язання оптимізаційних задач [2], метод з переходом від прямої задачі критеріального програмування до двоїстої) дозволяє користувачеві провести дослідження кожного із запропонованих методів і за виявом їхніх іманентних властивостей, особливостей, умов використання запропонувати власну оптимальну методику розв'язання конкретної задачі, яка завбачатиме застосування декількох наявних методів і матиме найкращі характеристики за кількістю ітерацій і точністю результатів розрахунку. У розрахункових програмах ПК ПОР передбачено графічну ілюстрацію розв'язку для візуального контролю збіжності і точності методів.

Такий підхід активізує творчі здібності студентів, заохочує їх до дослідницької роботи, змушує експериментувати, аналізувати, працювати самостійно. Лабораторно-практична робота в середовищі ПК ПОР вимагає відмови від інваріантних штампів щодо досліджуваного артефакту, вимагає розгляду та аналізу всіх характерних властивостей методів для вибору можливих способів розв'язання задачі.

Таким чином, суміщення індивідуального і самостійного навчання дозволяє користувачеві регулювати темп навчання і обирати його спрямування. Користувач може вибрати для вивчення ті чи інші модулі, замінивши відомі на нові для себе. Викладачеві надається можливість формувати окремі модулі на базі існуючих форм навчаючого та тестового матеріалу і створювати нові такі форми. З цією метою в програмному комплексі передбачено програму проектування навчаючих та тестових модулів.

Проектування навчаючих та тестових модулів.

Програма проектування модулів слугує основою для навчаючої і тестової програм, оскільки саме з її допомогою організуються навчаючі та тестові форми, з яких складаються окремі модулі. Програма реалізована мовою Object Pascal у візуальному середовищі Delphi.

Програмна реалізація середовища розробки навчаючих і тестових форм будується на ґрунті нового формату файлів з системою команд, які розпізнаються програмою, оскільки використання існуючих форматів (графічних файлів анімації, rtf-формату чи формату файлів довідки) зумовлює труднощі у створенні анімації та вимагає великого обсягу пам'яті. Зважаючи на те, що використання Методу-поля БД для зберігання навчаючих і тестових модулів виявилось неефективним, останні записуються в окремі текстові файли. Для їх координації та встановлення взаємозв'язків між ними створено базу даних такої структури: NUM — номер навчаючого чи тестового модуля; NAME — його назва; FILE — назва відповідного файлу. Для організації переходу між навчаючими модулями наявні поля LEVEL1, LEVEL2 — номери наступних модулів при першому та другому рівнях складності, COUNT — число затримок інформації на екрані, що необхідно для переходу до попереднього навчаючого модуля, IMAGE — назва фонового рисунка навчаючого модуля. Для визначення рівня складності тестового модуля використовується поле LEVEL3.

Програма проектування навчаючих і тестових модулів дозволяє змінювати записи бази даних та відповідні їм текстові файли (навчаючі і тестові модулі). Вікно проектування навчаючих і тестових модулів подано на рис.1.

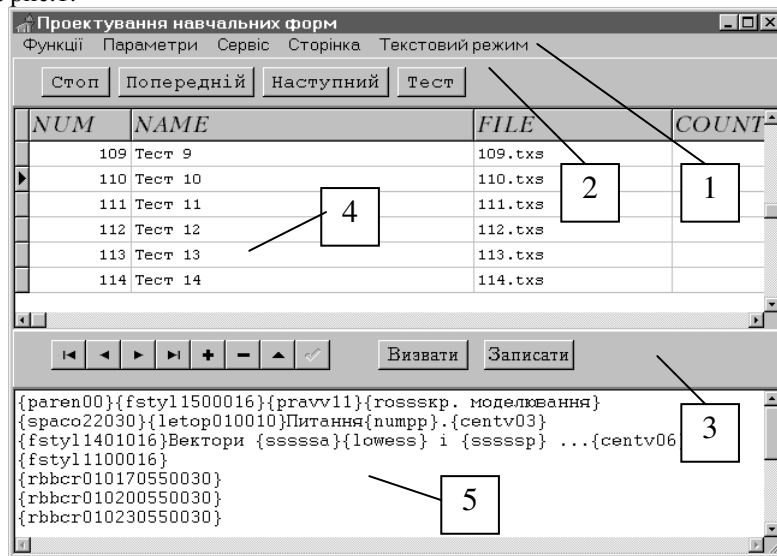


Рис. 1. Головне вікно програми проектування

Головне меню 1, панелі інструментів 2, 3 разом з реєстратором навчаючих модулів 4 і редактором форм 5 призначені для створення та редагування навчаючих і тестових форм, встановлення параметрів роботи програм. Текст навчаючих модулів складається з команд типу {спамераg}, структура яких така: спате — 5 символів назви команди, раg — параметри, кількість символів, зміст яких визначається командою. Команда може не включати параметри. Команда ідентифікується символом "{", а ознакою її завершення є символ "}".

Розглянемо основні команди проектування навчаючих і тестових форм.

Для затримки потоку інформації на **ab** секунд уведена команда {timera**ab**}, а затримку студювання інформації на час, визначений кількістю поданого матеріалу, — команда {times}.

Для стандартизації і спрощення в ході ілюстрації навчаючого матеріалу використовуються 2 шрифт Symbol — для відображення формул, — та Kudriashov — для відображення текстової інформації. Для встановлення параметрів шрифту використовується команда {fstyla**abvгдєє**}, тут **a** = 0, 1 — назва,

$\mathbf{b} = 0, 1, \dots, 9$ — колір, $\mathbf{v}, \mathbf{g}, \mathbf{d} = 0, 1$ — жирність, курсив, підкресленість, $\mathbf{e\text{e}}$ — розмір шрифту. Для зміни розміру шрифту існує команда $\{\mathbf{fonta\mathbf{b}}\}$, в якій параметр $\mathbf{a\mathbf{b}}$ — розмір шрифту.

Команда $\{\mathbf{s\text{расо}\mathbf{a\mathbf{b}\mathbf{v}\mathbf{g}\mathbf{d}}}\}$ встановлює постійний відступ зліва $\mathbf{a\mathbf{b}\mathbf{v}}$ та міжрядковий інтервал $\mathbf{g\mathbf{d}}$, а для задання поточного значення відступів зліва і зверху використовується команда $\{\mathbf{!e\text{тo}\mathbf{r}\mathbf{a}\mathbf{b}\mathbf{v}\mathbf{g}\mathbf{d}\mathbf{e}}\}$. Програма оперує набором 16 команд горизонтального і вертикального центрування.

Текст, який зчитується з навчаючих модулів зображається у вигляді створених міток (TLabel) на головному екрані чи на панелях (TPanel) подання інформації. Інформація може ілюструватися як на основному екрані так і на створених панелях, яким присвоюються порядкові номери їх створення. Переключення активної панелі виводу інформації здійснюється командою $\{\mathbf{pa\text{гe}\mathbf{n}\mathbf{a}\mathbf{b}}\}$, де $\mathbf{a\mathbf{b}}$ — номер створеної панелі; якщо $\mathbf{a\mathbf{b}} = "00"$, то як панель виводу інформації виступає головний екран.

Парцельовані конструкції тексту навчаючого модуля дозволяють уточнити окремі вирази за допомогою гіперпосилання у файл довідки текстового режиму. Такі вирази в тексті виділяються червоним кольором, курсивом і підкресленням. При підході до них курсор змінює форму (див. рис. 2) і програма дає можливість користувачеві здійснити вибір даного гіперпосилання. Це реалізує команда $\{\mathbf{!e\text{л}\mathbf{p}\mathbf{r}\mathbf{o}\mathbf{a}\mathbf{b}\mathbf{v}\dots}\}$, яка встановлює гіперпосилання останньої мітки на сторінку з назвою $\mathbf{"a\mathbf{b}\mathbf{v}\dots"}$.

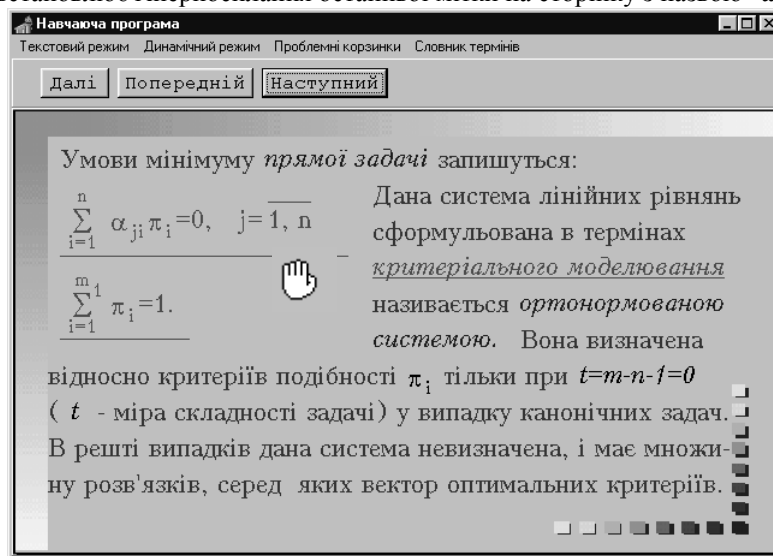


Рис. 2. Ілюстрація навчаючого модуля з гіперпосиланням.

Програмою передбачено широкий набір команд, які реалізують доступ до різних елементів (міток, панелей, кнопок відзначення тощо). Це команди створення, позиціонування, видимості елементів, зміни кольору міток, зміни напису кнопок, тощо.

Для написання формул використовуються додаткові команди: $\{\mathbf{!o\text{w}\mathbf{u}\mathbf{r}\mathbf{a}\mathbf{b}\mathbf{v}\dots\#\mathbf{гдe}\dots}\}$ — напис у верхньому і нижньому регістрах, $\mathbf{a\mathbf{b}\mathbf{v}\dots}$ — напис у нижньому регістрі; $\mathbf{гдe}\dots$ — напис у верхньому регістрі; $\{\mathbf{!m\text{u}\mathbf{l}\mathbf{e}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots\#\mathbf{гдe}\dots\#\mathbf{с\mathbf{ж}\mathbf{з}\dots}\}$, $\{\mathbf{!s\text{u}\mathbf{m}\mathbf{e}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots\#\mathbf{гдe}\dots\#\mathbf{с\mathbf{ж}\mathbf{з}\dots}\}$ — запис формули добутку та суми шрифтом Kudriashov; $\{\mathbf{!m\text{u}\mathbf{l}\mathbf{s}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots\#\mathbf{гдe}\dots\#\mathbf{с\mathbf{ж}\mathbf{з}\dots}\}$, $\{\mathbf{!s\text{u}\mathbf{m}\mathbf{s}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots\#\mathbf{гдe}\dots\#\mathbf{с\mathbf{ж}\mathbf{з}\dots}\}$ — запис формули добутку та суми шрифтом Symbol, тут $\mathbf{a\mathbf{b}\mathbf{v}\dots}$ — напис знизу; $\mathbf{гдe}\dots$ — напис зверху; $\mathbf{с\mathbf{ж}\mathbf{з}\dots}$ — напис справа; $\{\mathbf{!s\text{s}\mathbf{s}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots}\}$ — запис шрифтом Symbol; $\{\mathbf{!o\text{w}\mathbf{e}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots}\}$ — запис в нижньому регістрі; $\{\mathbf{!o\text{w}\mathbf{s}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots}\}$ — запис у нижньому регістрі шрифтом Symbol; $\{\mathbf{!u\text{p}\mathbf{p}\mathbf{e}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots}\}$ — запис у верхньому регістрі; $\{\mathbf{!u\text{p}\mathbf{s}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots}\}$ — запис у верхньому регістрі шрифтом Symbol, тут $\mathbf{a\mathbf{b}\mathbf{v}\dots}$ — текст напису.

З метою створення графічних модулів у редакторі форм передбачено команди зображення прямокутника, овалу, лінії і т.п. Графічний матеріал може бути також внесено у фоновий рисунок (назва вказується в полі IMAGE бази даних) навчаючого модуля.

Для реалізації тестових завдань до набору функцій додано низку команд, серед них: команда внесення в тестове завдання порядкового номера питання тестування $\{\mathbf{!n\text{u}\mathbf{m}\mathbf{p}\mathbf{r}\mathbf{r}}\}$, команда правильної відповіді на запитання $\{\mathbf{!p\text{r}\mathbf{a}\mathbf{v}\mathbf{v}\mathbf{a}\mathbf{b}\mathbf{v}\mathbf{g}\mathbf{d}\dots}\}$, тут \mathbf{a} — тип запитання, при $\mathbf{a} = 1$ — одна правильна відповідь (диз'юнктивний тип), \mathbf{b} — номер правильної відповіді, частина $\mathbf{v\mathbf{g}\mathbf{d}\dots}$ відсутня, при $\mathbf{a} = 2$ — декілька правильних відповідей (кон'юнктивний тип), \mathbf{b} — кількість правильних відповідей, $\mathbf{v}, \mathbf{g}, \mathbf{d}, \dots = 0, 1$ — правильність відповіді; команда назви розділу запитання $\{\mathbf{!o\text{r}\mathbf{s}\mathbf{s}\mathbf{s}\mathbf{a}\mathbf{b}\mathbf{v}\dots}\}$, тут $\mathbf{a\mathbf{b}\mathbf{v}\dots}$ — назва розділу. Перед записом на жорсткий диск файл тестового модуля (розширення txs) зашифровується. На рис. 3. подана ілюстрація тестового модуля.

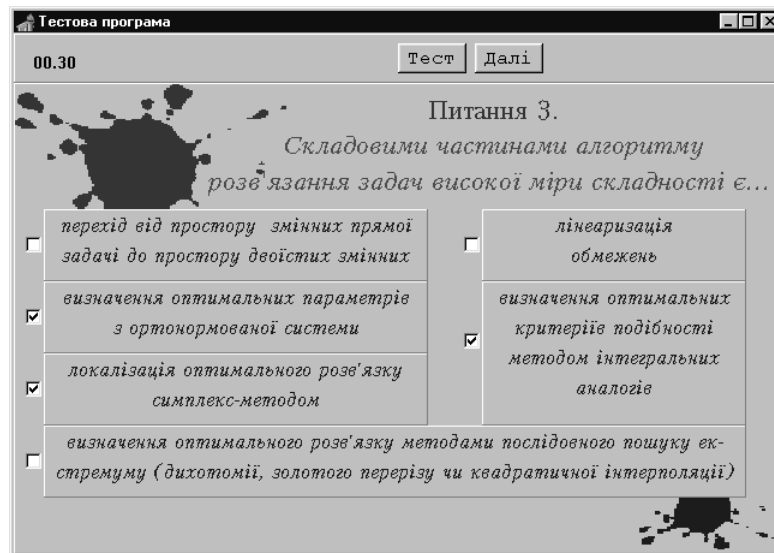


Рис. 3. Ілюстрація тестового модуля.

Для редагування та введення параметрів команд у зручному для користувача режимі в програмі реалізовано діалогове вікно, наприклад, для функції зміни поточного шрифту (рис. 4). Уводячи нову команду в навчаючий чи тестовий модуль, викликаємо діалогове вікно. Програма самостійно записує назву команди і значення параметрів за потрібним форматом у позицію курсору.

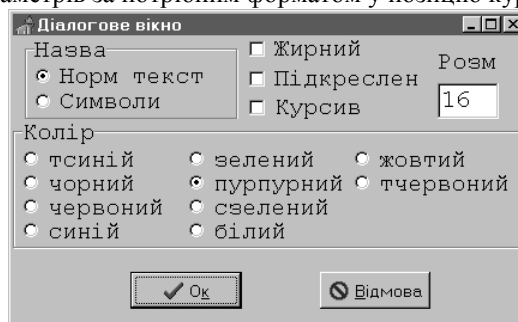


Рис. 4. Діалогове вікно установки параметрів команди зміни поточного шрифту

Доступ до програми проектування навчаючих і тестових форм обмежується паролем, який необхідно ввести в діалоговому вікні перед початком роботи з нею.

Висновки.

1. Досвід використання навчаючої та тестової програми в ПК ПОР протягом декількох років показав, що студенти виявляють значну зацікавленість щодо вивчення того матеріалу, який подається засобами навчаючої і тестової програми.

2. Використання розрахункової програми в ПК ПОР дозволяє поєднати елементи теоретичного і практичного характеру, на основі обчислювального експерименту перевірити працездатність моделей, гіпотез, теорій, зекономити час на проведення обчислювальних операцій.

3. Програма проектування навчаючих модулів дозволяє подати для вивчення і тестування будь-який матеріал. При цьому розробка навчаючих модулів може здійснюватися в будь-якому текстовому редакторі чи в програмі проектування навчаючих і тестових форм. Основною вимогою постає наявність у них команд, які розширюють можливості щодо подання інформації в динамічному режимі роботи навчаючої програми ПК ПОР.

4. Використання програми проектування при створенні навчаючих і тестових модулів дозволяє зменшити час, підвищити їх якість, залучити до цієї роботи розробників, які не мають спеціальної підготовки програміста і не знайомі з можливостями візуального середовища розробки Delphi.

Література.

1. Лежнюк П.Д., Бевз С.В. Методи оптимізації в електроенергетиці. Критеріальний метод. Навчальний посібник. - Вінниця: ВДТУ, 1999. — 177 с.

2. Бевз С.В. Адаптація критеріального методу до розв'язання поліноміальних задач в оптимальному керуванні // Книга за матеріалами V Міжнар. н.-т. конф. "Контроль і управління в технічних системах". — Вінниця. — 1999. — С. 179-185.