

## Оболонка CLIPS як засіб вивчення експертних систем

**Постановка проблеми.** Робота з програмним інструментарієм – один із головних напрямків сучасної шкільної інформатики, оскільки він дає можливість ознайомити учнів з інформаційними технологіями, з можливими сферами їх застосування в навчанні та в майбутній професійній діяльності. Традиційно вивчення теми “Штучний інтелект. Експертні системи” починається з подання прикладів штучного інтелекту, бо історично першим науковим напрямком, який обумовив появу самого терміну “штучний інтелект”, була імітація на ЕОМ творчих процесів: складання віршів і музики, доведення теорем, гра в шахи та шашки, переклад текстів тощо [1].

Сучасні експертні системи (ЕС) – це складні програмні комплекси, що акумулюють знання фахівців у конкретних предметних галузях і поширюють цей емпіричний досвід для консультування інших користувачів. Розробка експертних систем спрямована на використання ЕОМ для опрацювання даних в тих галузях науки і техніки, де традиційні методи моделювання малоприменні.

Саме тому при розгляді теми “Штучний інтелект. Експертні системи” головну увагу приділено готовим програмним засобам – оболонкам експертних систем FirstClass та Visual Expert. Головною особливістю цих оболонок є візуальний інтерфейс до бази знань та жорстко зафіксовані правила виведення. Це, з одного боку, дозволяє швидко побудувати експертну систему, але з іншого, – звужує клас систем, що можуть бути побудовані.

Візуальні оболонки ЕС уособлюють у собі найвищий рівень абстракції, тоді як функціональні та декларативні мови програмування – найнижчий. Середній рівень абстракції забезпечують оболонки ЕС, що, поряд із візуальним інтерфейсом, мають власні мови для створення баз знань та побудови запитів до них.

Останні оболонки є цілком придатними для навчання ЕС у ВНЗ як педагогічного, так і технічного спрямування: лише трохи програючи у простоті роботи суто візуальним оболонкам, вони надають можливості глибше опанувати розглядувану тему. При виборі засобу навчання ЕС доцільно застосувати табл. 1.

Табл. 1. Вибір засобу навчання ЕС в залежності від навчального часу

Навчальний час, год.	Середовище навчання ЕС	Рівень абстракції	Засіб навчання
< 6	Суто візуальна оболонка (найвищий рівень абстракції)	найвищий	FirstClass, Visual Expert
6–12	Візуальна оболонка із вбудованою мовою (середній рівень абстракції)	середній	CLIPS, BESS
> 12	Функціональна чи декларативна мова програмування	низький	Лісп, Пролог, Scheme

Серед багатьох існуючих оболонок ЕС однією з найбільш популярних є оболонка CLIPS, що має понад 20 років історії розвитку та широку інсталяційну базу [5]. Ця оболонка вигідно відрізняється від інших швидкодією та відкритістю, проте для не англійськомовного користувача її зручність обмежена неможливістю подання фактів та правил рідною мовою.

З метою подолання цього обмеження нами була розроблена локалізована версія CLIPS 6.23 (рис. 1), що може експлуатуватися на платформах Linux, DOS (лише текстове ядро), Windows і MacOS.

### 1. Основні характеристики оболонки ЕС CLIPS

Спочатку абревіатура CLIPS була назвою мови (мова Сі, інтегрована з продукційними системами), зручної для розробки баз знань і макетів експертних систем. Мова CLIPS була створена в Центрі космічних досліджень NASA у 1984 році.

Сьогодні CLIPS являє собою сучасний інструмент, призначений для створення експертних систем. CLIPS складається з керованого середовища – експертної оболонки зі своїм способом подання знань, гнучкої і потужної мови і кількох допоміжних інструментів.

Головна перевага CLIPS у тому, що мова і середовище CLIPS дають змогу користувачам швидко створювати ефективні, компактні і легко керовані експертні системи. При цьому користувач застосовує арсенал вже готових інструментів (механізм управління базою знань, механізм логічного виведення, менеджери різних об'єктів CLIPS) і конструктори (упорядковані факти, шаблони, правила, функції, родові функції, класи, модулі, вбудовану мову COOL для об'єктно-орієнтованого програмування).

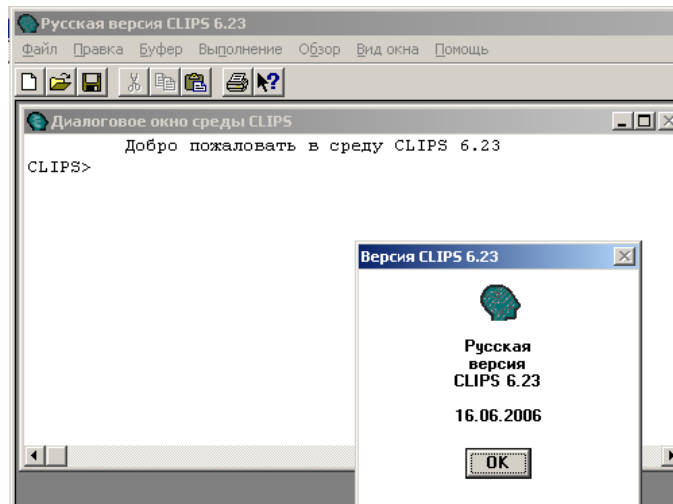


Рис. 1. Головне вікно CLIPS

Нами було локалізовано інтерфейс користувача, системні повідомлення та синтаксис CLIPS, що дає наступні переваги:

- а) зручність у використанні та при написанні експертних систем;
- б) якщо виникає помилка, система видає повідомлення рідною мовою, що полегшує розуміння змісту помилки;
- в) при написанні програми користувач має можливість застосовувати російські та українські позначення фактів, змінних, правил, процедур тощо.

Для демонстрації прикладів можна використовувати російську Windows-версію CLIPS 6.23, що цілком сумісна з базовою специфікацією мови.

Основним методом роботи з CLIPS є застосування командного рядка. Після появи в головному вікні CLIPS запрошення – *CLIPS>* – команди користувача можуть вводитися в середовище безпосередньо з клавіатури. Команди можуть бути викликами системних функцій чи функцій користувача, конструкторами різних даних CLIPS тощо. У випадку виклику користувачем деякої функції вона негайно виконується, а результат її роботи відображається користувачеві.

Всі дії, що виконуються у командному рядку, відображені у відповідних пунктах меню.

Для того щоб запустити експертну систему на виконання, необхідно виконати команду *reset* і команду *run*. Після цього можна з системою рацювати. Для повторного запуску експертної системи необхідно ще раз виконати команди *reset* і *run*.

Відразу після запуску середовища CLIPS на виконання на екрані з'являється запрошення, через яке сповіщається користувачеві, що він працює з інтерпретатором. У режимі інтерпретатора користувач може використовувати безліч команд: створювати нові факти, правила, описувати функції, використовувати конструктори, об'єкти CLIPS.

## 2. Принципи локалізації оболонки ЕС CLIPS

З метою локалізації CLIPS було необхідно:

- локалізувати інтерфейс користувача;
- локалізувати файли допомоги;
- локалізувати ядро системи.

Для локалізації інтерфейсу користувача були змінені наступні програмні ресурси: управління, меню, таблиці рядків, "гарячі" клавіші, шрифти.

Для локалізації файлів допомоги (рис. 2) було виконано їх переклад російською мовою з компіляцією у Help Workshop.

Для того, щоб встановити мову програми, яка впливає на використання функцій класифікації символів, у програмі явно вказано кодову сторінку операційної системи (для розглядуваної версії це "*Russian\_Russia.1251*").

Дійсною проблемою при локалізації завжди є операції з різними наборами символів. Існують такі мови і системи писемності, наприклад, японські ієрогліфи, у яких стільки знаків, що одного байта, за допомогою якого можна кодувати не більше 256 символів, просто недостатньо. Для підтримки подібних мов були створені двобайтові набори символів (символ представляється або одним, або двома байтами).

Компілятор ресурсів генерує двійкове представлення всіх ресурсів, використовуваних програмою. Рядки в ресурсах (таблиці рядків, шаблони допоміжних вікон, меню й ін.) завжди записуються в Unicode. В системі Unicode присвоюється унікальний код будь-якому символу, незалежно від платформи, незалежно від програми, незалежно від мови. Саме ця схема кодування використовується в таких сучасних технологіях і стандартах, як наприклад XML, Java, ECMAScript (JavaScript), LDAP, CORBA 3.0, WML і так далі. Саме Unicode є офіційною схемою реалізації ISO/IEC 10646. Нарешті, це кодування підтримується в багатьох операційних системах, в усіх сучасних браузерів Інтернет і у великій кількості інших програм.

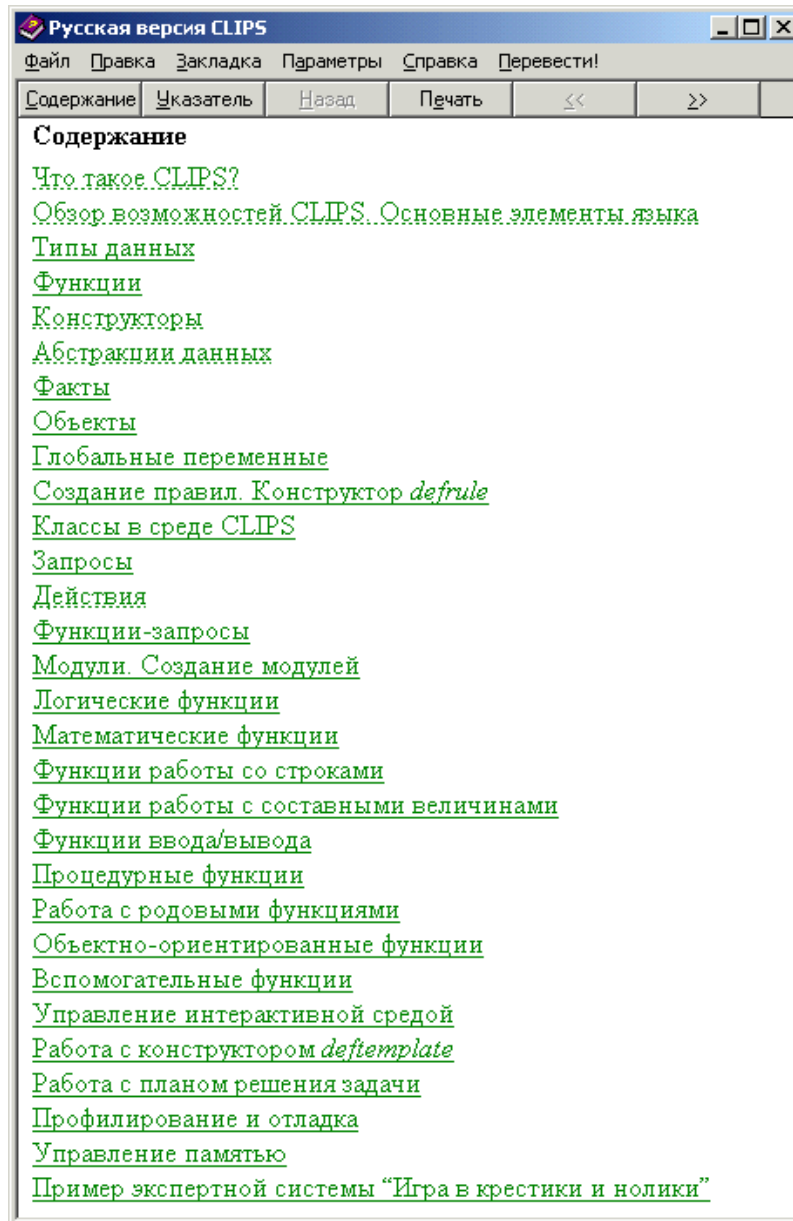


Рис. 2. Головной файл помощи

### 3. Интерфейс локализованой оболочки ЕС CLIPS

При запуске экспертной системы CLIPS появляется основное окно программы (рис. 1). Меню "Файл" (рис. 3) используется для работы с файлами данных:

- команда *Создать* – служит для создания нового документа;
- команда *Открыть* – команда открывает существующий файл;
- команда *Загрузка* – команда загружает конструкторы из текстового файла;
- команда *Загрузка командного файла* – загрузка командного файла;
- команда *Загрузка бинарного файла* – загрузка бинарного файла;
- команда *Загрузка текстового файла* – выполняет загрузку текстового файла в окно CLIPS;
- команда *Закрыть* – служит для закрытия документа;
- команда *Сохранить* – служит для поточного записывания конструкторов в текстовый файл;
- команда *Сохранить как* – служит для записывания текстового файла под новым именем;
- команда *Сохранить бинарный файл* – служит для записывания бинарного файла;
- команда *Возврат* – возвращает последнюю сохраненную версию файла;
- команда *Параметры страницы* – изменяет опции печати;
- команда *Печать* – печатает документ;
- команда *Выход* – выход из программы.

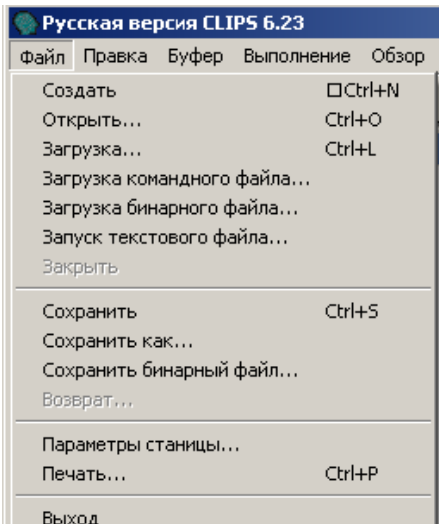


Рис. 3. Меню “Файл”

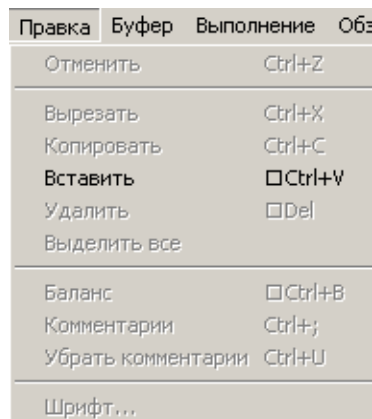


Рис. 4. Меню “Правка”

Меню “Правка” (рис. 4) використовується для редагування текстів програми:

- команда *Отменить* – слугує для відміни дій;
- команда *Вырезать* – для вирізання виділених областей тексту і надсилання їх до буферу обміна;
- команда *Копировать* – слугує для копіювання тексту;
- команда *Вставить* – для вставляння із буферу текстів програми;
- команда *Удалить* – для вилучення виділеного тексту;
- команда *Выделить все* – слугує для виділення всього тексту;
- команда *Баланс* – вибір тексту між дужками;
- команда *Комментарии* – слугує для встановлення коментарів після кожного символу рядка;
- команда *Убрать комментарии* – слугує для вилучення коментарів;
- команда *Шрифт* – служить для вибору шрифту.

Меню “Буфер” (рис. 5) використовується для заповнення буфера, пошуку даних:

- команда *Найти* – служить для пошуку тексту;
- команда *Заменить* – замінює наявний текст новим;
- команда *Загрузить выбранное* – служить для завантаження установок;
- команда *Выполнить выбранное* – здійснює установлені завантаження;
- команда *Загрузка буфера* – служить для завантаження буферу.

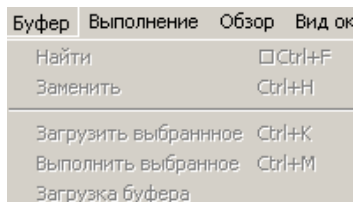


Рис. 5. Меню “Буфер”

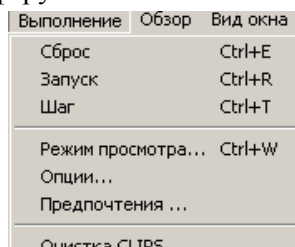


Рис. 6. Меню “Выполнение”

Меню “Выполнение” (рис. 6) використовується для компіляції даних, дозволяє встановлювати опції, очищає середовище:

- команда *Сброс* – служить для оновлення середовища;
- команда *Запуск* – служить для запуску програми на виконання;
- команда *Шаг* – покрокове виконання програми;
- команда *Режим просмотра* – дозволяє переглядати списки елементів середовища;
- команда *Опции* – дозволяє змінювати опції;
- команда *Предпочтения* – змінює всі дані;
- команда *Очистка CLIPS* – служить для очистки робочої пам’яті системи.

Меню “Обзор” (рис. 7) містить послуги для перегляду результатів виконання дій, правил, фактів, шаблонів, об’єктів, глобальних змінних, родових функцій та ін.:

- команда *Модуль* – включає в себе функцію main;
- команда *Менеджер правил* – служить для управління правилами;
- команда *Менеджер фактов* – служить для управління фактами;
- команда *Менеджер шаблонов* – служить для управління шаблонами;
- команда *Менеджер функций* – служить для управління функціями;
- команда *Менеджер глобальных переменных* – служить для управління глобальними змінними;
- команда *Менеджер классов* – служить для управління певними класами;
- команда *Менеджер предопределенных объектов* – служить для управління об’єктами;
- команда *Менеджер плана решений задачи* – служить для управління певними діями над активними правилами.

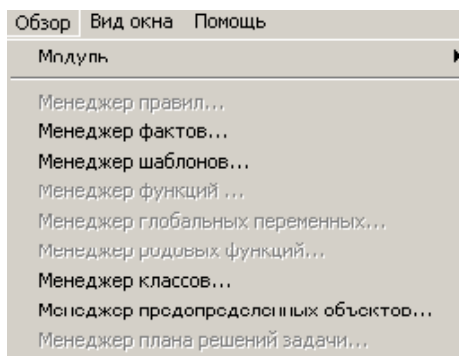


Рис. 7. Меню “Обзор”

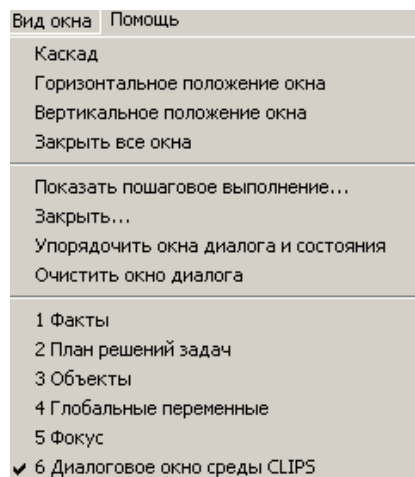


Рис. 8. Меню “Вид окна”

Меню “Вид окна” (рис. 8) використовується для зміни положення вікна середовища:

- команда *Каскад* – розміщує вікна каскадом;
- команда *Горизонтальное положение окна* – розміщує вікна у горизонтальному положенні;
- команда *Вертикальное положение окна* – розміщує вікна у вертикальному положенні;
- команда *Закрыть все окна* – зачиняє всі активні вікна середовища;
- команда *Пошаговое выполнение* – служить для показу покрокового виконання програми;
- команда *Закрыть* – зачиняє всі вікна покрокового виконання;
- команда *Упорядочить окна и состояния* – впорядковує всі вікна середовища;
- команда *Очистить окна* – очищає вікно від записів.

Меню “Помощь” (рис. 9) використовується для отримання довідкової інформації про роботу програми:

- команда *Версия CLIPS* – виводить відомості про версію CLIPS;
- команда *Справка по CLIPS* – виводить необхідні та додаткові відомості про CLIPS;
- команда *Пример экспертной системы* – виводить приклад експертної системи з коментарями;
- команда *Пример экспертной системы 2* – виводить приклад експертної системи з коментарями;
- команда *Строка состояния* – показує додаткові відомості.

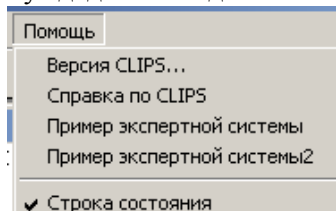


Рис. 9. Меню “Помощь”

#### 4. Приклад: експертна система “The Medic Diagnosis Expert System”

В меню “Помощь” наведена програма експертної системи, для діагностування загального стану хворого, визначення типу захворювання і надання користувачеві рекомендацій щодо надання першої медичної допомоги.

*Факти* – одна з основних форм подання повідомлень в CLIPS. У CLIPS фактом є список неподільних значень примітивних типів даних. CLIPS підтримує два типи фактів – упорядковані факти і неупорядковані факти чи шаблони. Факти можна додавати, вилучати, змінювати і дублювати, вводячи відповідні команди з клавіатури, або з програми.

*Правила* в CLIPS призначені для визначення евристик так званих “емпіричних правил”, що визначають набір дій, які виконуються при виникненні деякої ситуації. Правила (продукції) складаються з умов (Якщо) і наслідку (То). Ліва частина правила задається набором умовних елементів, що звичайно складаються з умов. Всі умови в лівій частині правила поєднуються за допомогою неявного логічного оператора *and*. Права частина правила містить список дій, виконуваних при використанні правила в результаті застосування механізму логічного висновку. Для поділу правої і лівої частини правил використовується символ  $\Rightarrow$ . Дії правила виконуються послідовно, але тоді і тільки тоді, коли всі умовні елементи в лівій частині цього правила задовольняють іншим. Якщо в правій частині правила не визначена жодна дія, правило може бути застосоване, але при цьому нічого не відбудеться.

*Функцією* в CLIPS називається частина коду, що має ім'я і повертає результат виконаної дії (наприклад, зображення повідомлень на екрані). Функції, що не повертають результат і при цьому виконують роботу, називаються командами. CLIPS оперує з кількома типами функцій – визначені користувачем *зовнішні функції*, *системні (внутрішні) функції*, функції, визначені в середовищі CLIPS за допомогою конструктора *deffunction* та *родові функції*.

Функції можуть вводитися в вікно CLIPS з клавіатури або використовуватися в правилах, повідомленнях, визначених користувачем в функціях чи родових функціях. CLIPS надає великий набір функцій, за допомогою яких можна задовольнити будь-які потреби користувача, серед яких логічні і математичні функції, функції роботи з рядками і складеними величинами, функції введення/виведення, проце-



дурні функції, функції для роботи з методами родових функцій, функції для роботи з конструкторами.

Щоб запустити приклад ЕС, його слід зберегти в текстовому файлі з розширенням *.clp* та за допомогою команди *load* завантажити програму в середовище CLIPS. Далі треба виконати команди (*reset*) і (*run*). Після цього система повідомляє про наявність помилок у програмі, а якщо вони відсутні, то програма готова для демонстрації. Далі відбувається “спілкування” між системою і користувачем, при цьому вона видає певні рекомендації (рис. 10).

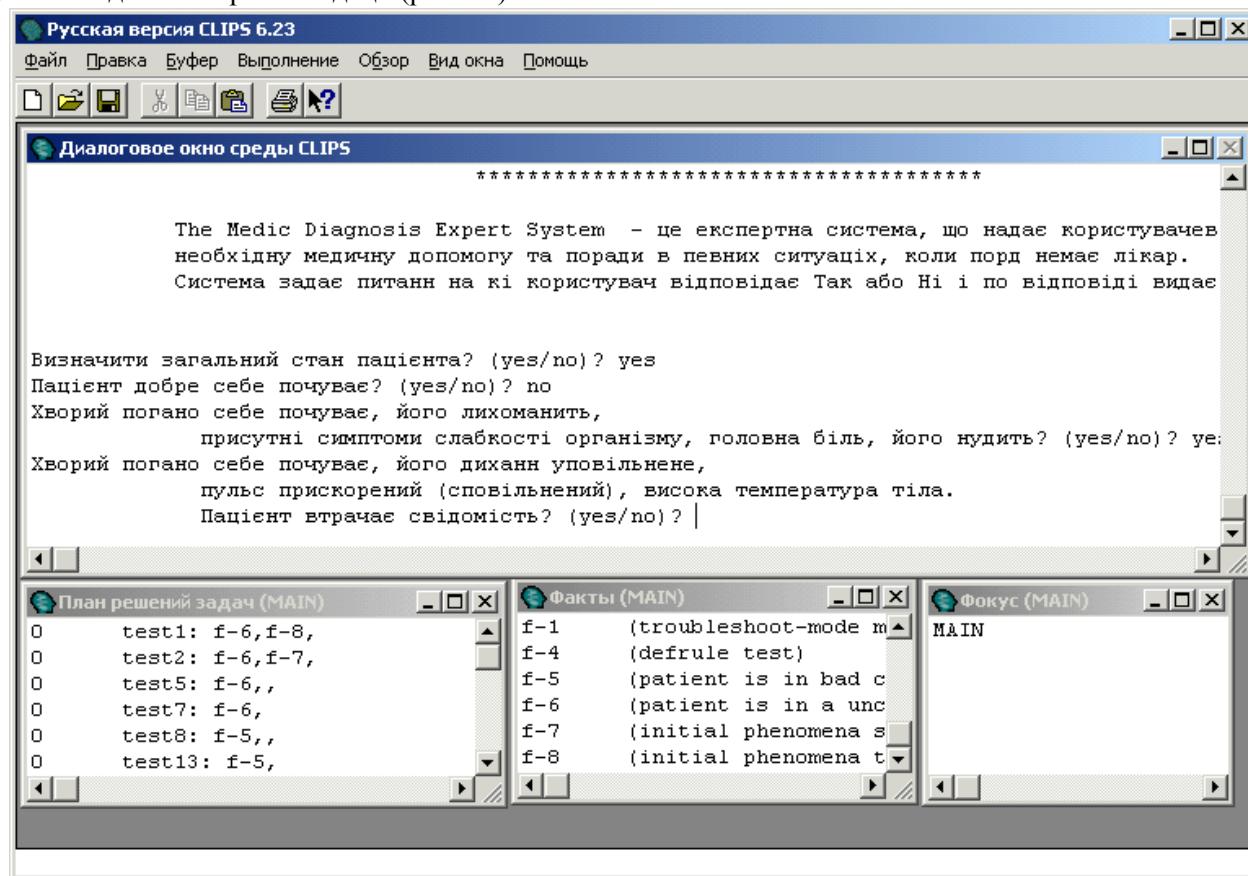


Рис. 10. Демонстрація експертної системи

## Висновки

1. В статті в якості інструментального середовища розробки експертних систем запропонована експертна оболонка CLIPS. Однією з основних характеристик CLIPS є її продуктивність, тобто швидкість одержання результату і його вірогідність (надійність). CLIPS "має змогу" пояснити, чому запропоновано саме таке рішення, і довести його обґрунтованість. Система не вимагає інсталяції, мала за розміром (не більше 825 кілобайт) та невимоглива до апаратних ресурсів.

2. Локалізація CLIPS передбачає локалізацію інтерфейсу користувача, ядра та синтаксису CLIPS. Застосування локалізованої версії дає наступні переваги:

- зручність у використанні та при написанні експертних систем;
- якщо виникає помилка, видається повідомлення рідною мовою, що полегшує розуміння змісту помилки;
- при написанні програми користувач має можливість застосовувати російські та українські позначення фактів, змінних, правил, процедур тощо.

3. CLIPS – це вільно поширюваний продукт, локалізована версія якого створена для всіх сучасних операційних систем. Це дає можливість стабілізувати засоби навчання та зміст теми “Штучний інтелект. Експертні системи”, зробивши її незалежними від операційної системи [2].

4. Усталення змісту та засобів навчання інформатики через уникнення їх залежності від операційної системи, мови програмування тощо, сприяє підвищенню рівня теоретичної підготовки, дозволяє створити стабільні підручники, надає широкі можливості для вибору апаратних та програмних засобів навчання інформатики, знижуючи вартість володіння ними за рахунок використання ліцензійно чистого вільно поширюваного програмного забезпечення [3].

## ЛІТЕРАТУРА

- Рамський Ю.С., Балик Н.Р. Методичні основи вивчення експертних систем у школі. – К.: Логос, 1997. – 128 с.
- Семеріков С.О., Теплицький І.О. Вільне програмне забезпечення як фактор стабілізації вузівських курсів інформатики / Інформаційні технології в освіті: Матеріали Всеукраїнської науково-практичної конференції (24–26 травня 2006 р.). – Мелітополь: МДПУ, 2006. – С. 55–56.
- Семеріков С.О., Теплицький І.О. Інваріантність до операційної системи та мови програмування як засіб фундаменталізації курсів інформатики у ВНЗ // Інформаційні технології в освіті, науці і техніці / Матеріали V Всеукраїнської конференції молодих науковців ІТОНТ–2006: Черкаси, 3–5 травня 2006 р. – Черкаси: ЧНУ, 2006. – С. 140.
- Частиков А.П., Гаврилова Т.А., Белов Д.Л. Разработка экспертных систем. Среда CLIPS. – СПб.:

БХВ-Петербург, 2003. – 608 с.: ил.