

стала фундаменталізація цього курсу. Мабуть, всім відома думка про те, що знання небагатьох принципів замінює знання багатьох фактів. Вона і є основою фундаменталізації. Учень, який оволодів принципами, легко освоїть різні програмні продукти, що побудовані на тих принципах. Важливо відмітити, що фундаменталізація багато в чому базується на інформаційному моделюванні, що, на нашу думку, повинно стати стрижнем вивчення інформатики. Саме метод моделювання дозволяє поєднати теоретичні засади інформатики з їх практичною реалізацією, наприклад теоретичну реляційну модель даних з SQL моделлю.

Фундаменталізація курсу інформатики спонукує демонструвати реалізацію одних і тих самих принципів у конкуруючих програмних засобів, що призводить до впровадження вільно поширюваного програмного забезпечення. Таке програмне забезпечення вже стало конкурентним до комерційного у навчальному процесі і в бізнесі. Так, зараз у Чернігівському педагогічному університеті на фізико-математичному факультеті студенти обов'язково вивчають операційну систему (ОС), основу на GNU/Linux, а MS Office повністю замінено на Libre Office. На нашу думку, перспективним є відхід від циркуляції документів у закритих форматах до відкритих форматів документів. Щодо накопичених ППЗ, орієнтованих на ОС Windows, використовувати при роботі з іншими ОС дозволяють технології емуляції та віртуалізації.

Тепер потрібно сказати про перспективи розвитку програмної складової ППЗ Gran1, може поки що досить далекі. А в перспективі нас можуть чекати досить серйозні зміни, пов'язані із змінами самої ІТ-індустрії, а, отже, і шкільної та вузівської інформатики. Все більш популярною стає думка про те, що індустрія ПК, яку ми знаємо вже багато років, починає згортатися. Цьому є 2 причини:

- 1) тотальне розповсюдження Інтернету, радикальне збільшення його швидкодії і значне зменшення плати за його використання;
- 2) обчислювальні характеристики смартфонів, планшетів або інших мініатюрних пристроїв наближаються до обчислювальних характеристик ПК.

Користувачі бажають мати можливість використовувати комп'ютер будь-де, а не тільки на робочому місці. В той же час виявилось, що настільні ОС не придатні для таких пристроїв, і ситуація, що складається на ринку, не дозволяє говорити про домінування якоїсь ОС, як це склалося на ПК з ОС Windows. Кросплатформенність, віртуалізація та емуляція для таких пристроїв – поганий вибір внаслідок обмежених обчислювальних потужностей, часу роботи від батареї і певної протидії розробників мобільних ОС. Переписувати ж програму для кожної ОС для колективу ентузіастів практично неможливо. Вихід можна бачити в хмарних обчисленнях, а саме в тій їх складовій, яка називається RIA (Rich Internet Applications, або Веб-застосунками: програма розміщується на сервері, а робота з нею відбувається через браузер (приклад – Google Docs). Це дозволить працювати з програмою за допомогою будь-якого пристрою з сучасним браузером. Зрозуміло, що перехід до цієї концепції у випадку Gran1 вимагає великого обсягу роботи і пошуків шляхів для його реалізації.

Широке застосування ППЗ Gran у навчальному процесі шкіл та ВНЗ України, Росії, Белорусі, Польщі тощо, а також численні дисертації, захищені з використанням програмного комплексу GRAN, свідчить про перспективність інформатизації навчального процесу, розробки і впровадження сучасних комп'ютерно-орієнтованих методичних систем навчання всіх без винятку предметів, в тому числі і математики та інформатики.

Список використаних джерел

1. Горошко Ю.В., Цибко Г.Ю. Методика навчання інформатики – історія і перспективи // Вісник Чернігівського державного педагогічного університету імені Т.Г.Шевченка. Випуск 93. Серія: педагогічні науки: Збірник. – Чернігів: ЧНПУ, 2011. – №93. – С.61-64.
2. Жалдак М.І., Горошко Ю.В., Вінниченко Є.Ф. Математика з комп'ютером. Посібник для вчителів. – К. : НПУ імені М.П. Драгоманова. 2009. – 282 с.
3. Жалдак М.І. Педагогічний потенціал комп'ютерно-орієнтованих систем навчання математики // комп'ютерно-орієнтовані системи навчання : зб. наукових праць. – Вип. 7 – К. : НПУ імені М.П. Драгоманова. – 2003. – С. 3-16.

Мінтій І. С., Семеріков С. О., Тарасов І. В.
ДВНЗ «Криворізький національний університет»

Методика формування у майбутніх учителів інформатики компетентностей з програмування на прикладі теми «Експертна система»

Нові галузеві стандарти вищої освіти України розроблено на основі компетентнісного підходу. Тому актуальним є пошук практичних шляхів реалізації цього підходу у вищій освіті, що є особливо важливим для підготовки вчителя інформатики у зв'язку з постійним розвитком науки інформатики та засобів ІКТ.

В статті обговорюються можливі підходи до формування у майбутніх учителів інформатики компетентностей з програмування на прикладі теми «Експертна система» курсу «Вступ до програмування» [2].

Тема «Експертні системи» віднесена до третього модуля даного курсу – «Практичні схеми». Робота над будь-якою темою з цього модуля виконується за кілька етапів:

- 1) ознайомлення з теоретичним матеріалом;
- 2) тестування програми та аналіз результатів;
- 3) модифікація та вдосконалення програми;
- 4) виконання колективних проектів без створення графічного інтерфейсу програми;
- 5) вдосконалення та захист колективних проектів з графічним інтерфейсом.

Так, наприклад, при опануванні матеріалу з даної теми на *третьому етапі* студенти отримують наступні завдання [1]: додати до експертної системи меню, що містить пункти:

1. Провести експертну оцінку.
2. Вивести відомості про
3. Пояснити останнє виведення.
4. Завершення роботи.

В результаті обрання першого пункту повинна запускатися машина виведення, другого – вводиться запит про предмет і виводиться всі правила і гіпотези, що його містять, третього – в зручній формі виводиться ланцюжок висновків стосовно обґрунтування останнього твердження (гіпотези) у вигляді послідовностей «оскільки ..., можна зробити висновок, що ...». Вибір четвертого пункту повинен призводити до завершення роботи з системою.

На *четвертому етапі* роботи над даною темою студенти отримують завдання: створити одну із експертних систем:

– експертну систему, за допомогою якої можна з'ясувати, яку формулу краще застосовувати при виконанні певного типу завдань під час навчання шкільного курсу фізики (розділи кінематика і динаміка). В якості гіпотези потрібно задати формули, правила побудувати на основі наступного фрагменту бази знань:

багатозначний(формула)

дозвзн(розділ) = кінематика, динаміка

дозвзн(рух_тіла) = прямолінійний, обертальний

дозвзн(швидкість) = постійна, рівнозмінна

дозвзн(причини_руху) = не_враховуються, враховуються

дозвзн(сила) = сила_тертя, сила_тяжіння, сила_пружності, кілька_сил

дозвзн(зад_сил) = задані, не_задані

питання(причини_руху) = Чи враховуються причини руху?

питання(рух_тіла) = Який характер руху тіла?

питання(швидкість) = Який характер швидкості?

питання(сила) = Яка сила розглядається в задачі?

питання(зад_сил) = Чи задані сили?

правило0:

якщо розділ = кінематика

то підрозділ = невизн.

правило1:

якщо причини_руху = не_враховуються

то розділ = кінематика.

правило2:

якщо причини_руху = враховуються

то розділ = динаміка.

правило3:

якщо розділ = кінематика

і рух_тіла = прямолінійний

і швидкість = постійна

то формула = $X=X_0+V*t$

і формула = $V=S/t$.

правило4:

якщо розділ = кінематика

і рух_тіла = прямолінійний

і швидкість = рівнозмінна

то формула = $a=(V-V_0)/t$
і формула = $V=V_0+a*t$
і формула = $X=X_0+V_0*t+A*t^2/2$
bmp = pryskor.

правило5:

якщо розділ = кінематика

і рух_тіла = обертальний

то формула = $w=f/t$

і формула = $V=w*r$

і формула = $a=V^2/r$.

правило6:

якщо розділ = динаміка

і зад_сил = задані

то підрозділ = рух_під_дією_сил.

правило7:

якщо розділ = динаміка

і зад_сил = не_задані

то підрозділ = закони_ньютонана.

правило8:

якщо розділ = динаміка

і підрозділ = закони_ньютонана

то формула = $F=M*a$

і формула = $F_1=-F_2$.

правило9:

якщо розділ = динаміка

і підрозділ = рух_під_дією_сил

і сила = сила_тертя

то формула = $F=-k*N$.

правило10:

якщо розділ = динаміка

і підрозділ = рух_під_дією_сил

і сила = сила_тяжіння

то формула = $F=G*M_1*M_2/r^2$.

– експертну систему «Класифікатор рослин» на основі наступного фрагменту бази знань:

Якщо клас голонасінні і форма листка лускоподібна, то сімейство – кипарисові.

Якщо клас голонасінні і форма листка голкоподібна і конфігурація хаотична, то сімейство – соснові.

Якщо клас голонасінні і форма листка голкоподібна і конфігурація – 2 рівних ряди і срібляста смуга, то сімейство – соснові.

Якщо клас голонасінні і форма листка голкоподібна і конфігурація – 2 рівних ряди і сріблястої смуги немає, то сімейство – болотний кипарис.

Якщо тип – дерева і форма листка широка і пласка, то клас – покритонасінні.

Якщо тип – дерева і не підтверджується, що форма листка широка і пласка, то клас – голонасінні.

Якщо стебло зелене, то тип – трав'янисті.

Якщо стебло дерев'янисте і стелиться, то тип – ліани.

Якщо стебло дерев'янисте і прямостояче і один основний стовбур, то тип – дерева.

Якщо стебло дерев'янисте і не підтверджується, що положення прямостояче і один основний стовбур, то тип – чагарникові.

– Створити експертну систему «Класифікатор птахів», за допомогою якої можна розрізнити орла, сокола і горобця, використовуючи в якості принципів класифікації ім'я об'єкта, довжину дзьоба, забарвлення пір'я, розмах крил, висоту польоту, аеродинамічні принципи.

– Створити експертну систему «Прогнозування повеней» на основі наступного фрагменту бази знань:

Якщо рівень-води = високий і дощ = рясний, то місто = евакуювати.

Якщо рівень-води = високий і дощ = не сильний і снігу = багато, температура = висока, то місто = евакуювати.

Якщо рівень-води = високий і дощ = не сильний і снігу = багато, температура = середня і дощ = помірний, то місто = посилити увагу.

Якщо рівень-води = високий і дощ = немає і снігу = багато, температура = середня і дощ = слабкий, то місто = не турбуватись.

Якщо рівень-води = високий і дощ = не сильний і снігу = мало, то місто = не турбуватись.

Якщо рівень-води = невисокий і дощ = сильний і снігу = багато, температура = висока, то місто = посилити увагу.

Якщо рівень-води = невисокий і дощ = сильний і снігу = багато, температура = середня, то місто = не турбуватись.

Якщо рівень-води = невисокий і дощ = сильний і снігу = мало, то місто = не турбуватись.

Якщо рівень-води = високий і дощ = несильний, то місто = не турбуватись.

– Створити експертну систему «Прийом на роботу», за допомогою якої реалізують схему, зображену на рис. 1.

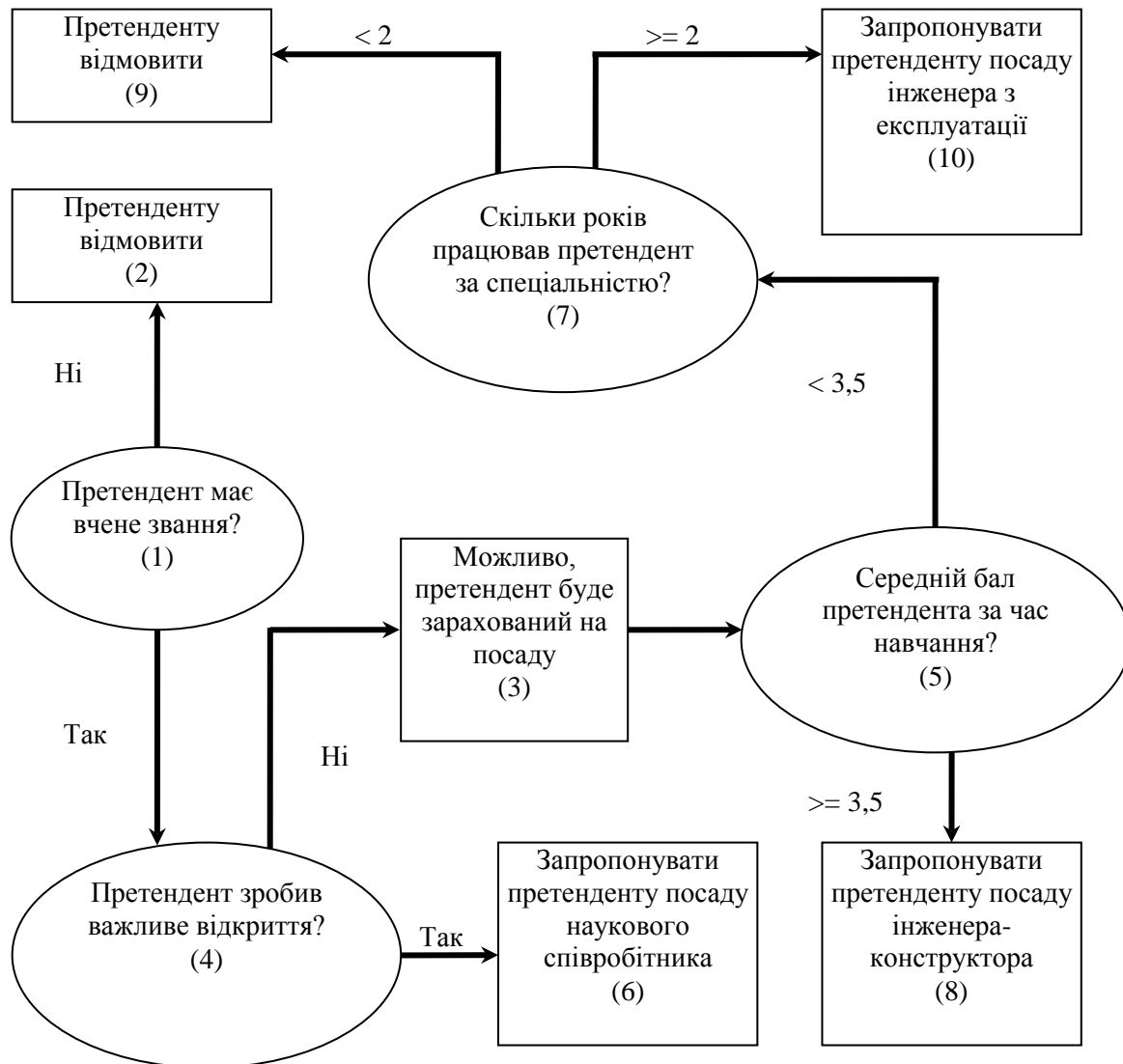


Рис. 1. Схема експертної системи «Зарахування-на-посаду»

6. Експертна система за власним вибором.

П'ятий етап роботи: після вивчення теми «Графічний інтерфейс» студенти отримують завдання розробити графічний інтерфейс до вже створеної ними експертної системи та публічно захищають виконані проекти.

У розробленому за допомогою електронної системи підтримки навчання Moodle навчально-методичному комплексі «Вступ до програмування» до всіх робіт з цього модуля розроблені такі матеріали: завдання проекту; робочий файл проекту; відео-приклад роботи над проектом; звіт з проекту.

Завдання проекту містять теоретичний матеріал та більшість функцій, необхідних для «базової» роботи над проектом (в наступних завданнях проект буде модифіковано й ускладнено).

Робочий файл проекту містить або програму, що повністю готова до роботи, або ж за виключенням деяких необхідних функцій. У відео-прикладі демонструється запуск проекту та можливі результати (рис. 2, рис. 3).

```

Мова: racket; memory limit: 128 MB.
> (Експерт)
Чи це правда, що: тварина має пір 'я
Так
Згідно правила ПРАВИЛО3: тварина птах
Чи це правда, що: тварина не вміє літати
так
Чи це правда, що: тварина плаває
так
Чи це правда, що: тварина чорно-біла
так
Згідно правила ПРАВИЛО14: тварина пінгвін
(тварина пінгвін)

```

Рис. 2. Розпізнавання пінгвіна

```

Мова: racket; memory limit: 128 MB.
> (Експерт)
Чи це правда, що: тварина має пір 'я
Ні
Чи це правда, що: тварина вміє літати
ні
Чи це правда, що: тварина має шерсть
Так
Згідно правила ПРАВИЛО1: тварина ссавець
Чи це правда, що: тварина має копита
так
Згідно правила ПРАВИЛО7: тварина жуйна
Чи це правда, що: тварина смугаста
так
Згідно правила ПРАВИЛО12: тварина зебра
(тварина зебра)
>

```

Рис. 3. Розпізнавання зебри

В процесі опанування матеріалу з даної теми передбачається підготовка відповідних повідомлень та виступ студентів на лекціях з доповідями:

- Бази знань. Моделі подання знань.
- Експертні системи. Технологія розробки баз знань із використанням експертних оболонок;
- Покоління експертних систем.

Цей вид роботи сприяє формуванню у студентів здатностей знаходити матеріал за темою доповіді, аналізу знайдених відомостей, формулюванню висновків, досвіду публічного захисту виконаної роботи.

Під час виконання проектів з даного модуля у студентів окрім систематизації, поглиблення та вдосконалення теоретичних знань і вмінь, одержаних під час опанування матеріалу попередніх модулів, формуються такі здатності, що характеризують сформованість компетентностей з програмування:

– *гносеологічна складова*: знання характеристик експертних систем; знання структури експертної системи та етапів її розробки; знання класифікацій експертних систем; знання відомих експертних систем; уявлення про моделі подання знань; уявлення про сфери застосування та перспективи розвитку експертних систем;

– *праксеологічна складова*: розуміння, налагодження та модифікація програм; бачення задачі одночасно на різних рівнях деталізації; узагальнення типових задач; використання знань і навичок в нових ситуаціях; розуміння зв'язків теорії програмування із практикою програмування; бачення галузі застосування здобутих знань; встановлення причинно-наслідкових зв'язків; створення документації до програми;

– *аксіологічна складова*: емоційно-ціннісне ставлення до процесу розробки, опису, налагодження, тестування програм та аналізу результатів їх застосування; знаходження нових, нестандартних рішень стосовно розв'язування задачі; внутрішня мотивація до опанування програмуванням; прагнення до самовдосконалення, потреба у саморозвитку гносеологічних та праксеологічних складових системи загальнокультурних та професійних компетентностей; самостійне прийняття рішень, критичне ставлення до різноманітних повідомлень і тверджень, здатність за власним почином організувати діяльність, ставити мету, в разі необхідності вносити в поведінку зміни; точність, дисциплінованість, акуратність, уважність, внутрішня керованість; цілеспрямованість, наполегливість, працездатність і ретельність у праці; внутрішня потреба у створенні програмних продуктів;

– *соціально-поведінкова складова*: здатність працювати як індивідуально, так і в колективі (як лідер або член колективу); розподіл функцій і взаємостосунків у колективі; точне і лаконічне формулювання та висловлювання власних думок і ідей; аргументоване відстоювання власної точки зору; погодження своїх міркувань з іншими учасниками колективу; аналіз власних помилок; розуміння та готовність до конструктивного обговорення думок та ідей інших; розуміння відповідальності за виконану роботу; використання засобів для організації спільної роботи над проектом.

Висновки. Під час вивчення теми «Експертні системи» відбувається формування усіх складових компетентностей з програмування – гносеологічної (через здобуття знань безпосередньо з даної теми), праксеологічної (через необхідність застосування вже набутих знань до розв'язування практично-значущих задач), аксіологічної та соціально-комунікативної (через використання методу проєктів).

Список використаних джерел

1. Мінтій І. С. Схематичне програмування (початки програмування: функціональний підхід) / І. С. Мінтій; за ред. академіка НАПН України М. І. Жалдака. – К. : НПУ ім. М. П. Драгоманова, 2010. – 152 с.

2. Мінтій І. С. Формування у студентів педагогічних університетів компетентностей з програмування на основі функціонального підходу : дис. ... канд. пед. наук : 13.00.02 – теорія та методика навчання (інформатика) / Ірина Сергіївна Мінтій; Національний педагогічний ун-т ім. М. П. Драгоманова. – 2013. – 254 с.

Балик Н.Р.

Тернопільський національний педагогічний університет імені Володимира Гнатюка

Деякі аспекти теоретико-методологічних основ формування професійної магістерської програми зі спеціальності «Інформатика» з присвоєнням кваліфікації «Викладач інформатики»

Анотація. В умовах реформування сучасної системи вищої освіти (Наказ МОН № 99 від 10.02.10 року) особливої актуальності набуває проблема розробки нових стандартів вищої освіти для освітньо-кваліфікаційного рівня «магістр» з нормативною тривалістю навчання 1,5-2 роки. Матеріали даної статті присвячені питанням аналізу структури змісту професійної магістерської програми зі спеціальності «Інформатика» з присвоєнням кваліфікації «Викладач інформатики», вимогам до формування змісту таких програм. Розглянуто навчальні плани магістерської підготовки у контексті формування варіативної її частини, описано окремі концепції, що стосуються теоретико-методологічних основ формування професійної магістерської програми зі спеціальності «Інформатика», перелічено вибірково курси для магістрантів, що читалися у Тернопільському національному педагогічному університеті імені Володимира Гнатюка.

Постановка проблеми та аналіз останніх досліджень. Висока динаміка сучасного ринку праці диктує необхідність орієнтації підготовки магістрів на його конкретні потреби. Навчання магістрів в Україні здійснюється за спеціальностями та освітніми програмами. Освітні програми підготовки магістрів поділяють на:

- *дослідницькі*, за якими передбачається поглиблення досліджень в одній з наукових галузей;
- *професійні*, за якими передбачається розвиток професійних та формування управлінських компетенцій у певній галузі професійної діяльності;
- *кар'єрні*, за якими передбачається вдосконалення здобутих теоретичних знань і практичного досвіду для кар'єрного зростання та підготовки до здійснення управлінської діяльності.

У концепції організації підготовки магістрів в Україні зафіксовано необхідність подовження нормативного терміну навчання магістрів з усіх напрямів до 1,5-2 років з метою створення належних умов для поєднання теоретичної та практичної їх підготовки (вже при півторарічній тривалості