

УДК 378.016:004.087

© Осадчий В.В., Осадча К.П.

ДОСВІД РОЗРОБКИ ТА АПРОБАЦІЇ НАВЧАЛЬНОГО КУРСУ «ВСТУП ДО СПЕЦІАЛЬНОСТІ» (ІНЖЕНЕР-ПРОГРАМІСТ)

Постановка проблеми. З кожним роком зростає зацікавленість випускників середньої школи професією програміст, тому що вона одна з найбільш затребуваних спеціальностей у сучасному суспільстві. Але далеко не всі уявляють, що потрібно знати й уміти, щоб стати висококваліфікованим спеціалістом в галузі програмування. Саме тому підготовка фахівців цього напрямку є актуальною проблемою, що дозволить підвищити рівень інформатизації суспільства та збільшити українську частину ринку програмних розробок. Підготовка програмістів в Україні відбувається у вищих навчальних закладах за напрямами підготовки 040302 "Інформатика" і 050103 "Програмна інженерія".

Підготовка фахівців за напрямом підготовки 040302 – Інформатика [3] дозволяє отримати такі кваліфікаційні рівні: бакалавр (строк навчання 4 роки, кваліфікація: фахівець з інформаційних технологій); спеціаліст (строк навчання 1 рік, кваліфікація: інженер-програміст); магістр (строк навчання 1 рік, кваліфікація: магістр з інформатики). За напрямом підготовки: 050103 – програмна інженерія [4]: бакалавр (строк навчання 4 роки, кваліфікація: технік-програміст); спеціаліст (строк навчання 1 рік, кваліфікація: інженер-програміст); магістр (строк навчання 1,5 роки, кваліфікація: магістр з програмного забезпечення автоматизованих систем).

Домінуючими сферами діяльності випускників цих спеціальностей є: консультування з питань створення програмного забезпечення та надання допомоги щодо технічних аспектів комп'ютерних систем; розроблення індивідуального програмного забезпечення (на замовлення) та адаптування пакетів програм до специфічних потреб користувачів; надання послуг із системного аналізу, програмування та супроводу, а також інших спеціалізованих послуг у сфері інформатизації, консультування щодо типу та конфігурації комп'ютерних технічних засобів та використання програмного забезпечення: аналіз інформаційних потреб користувачів та пошук оптимальних рішень тощо.

Можливі посади, за якими буде працевлаштований фахівець, це: інженер-програміст; фахівець з інформаційних технологій; фахівець із розробки та тестування програмного забезпечення; науковий співробітник; викладач ВНЗ.

Постановка завдання. Мета статті полягає в описі розробленого навчального курсу «Вступ до спеціальності» (інженер-програміст): його змісту, мети, завдань; характеристики друкованого посібника, електронного підручника та підсумкового тестового контролю, а також висвітлення результатів апробації курсу.

Виклад основного матеріалу. На факультеті інформатики, математики та економіки Мелітопольського державного педагогічного університету імені Богдана Хмельницького дисципліна «Вступ до спеціальності» викладається для студентів із галузі знань 0403 Системні науки та кібернетика, які отримують кваліфікацію «Фахівець з інформаційних технологій» на кваліфікаційному рівні «бакалавр» та «Інженер-програміст» на кваліфікаційному рівні «спеціаліст».

Відповідно до запитів інформаційного суспільства та змін, які відбуваються у професійній підготовці програмістів, виникла нагальна потреба в удосконаленні змісту дисципліни «Вступ до спеціальності». До її змісту ввійшли вимоги, які пред'являються програмістам на ринку праці та академічні засади їхньої професійної підготовки.

Аналіз навчальних планів дисципліни «Інформатика» загальноосвітніх шкіл ([1], [2], [7], [8]), показав недостатню кількість годин на вивчення алгоритмізації та основ програмування. Це призвело до формування низького початкового рівня підготовки майбутніх програмістів, їхньої недостатньої мотивації до навчання та нерозуміння змісту їхньої підготовки.

Це підтвердили дані нашого вхідного опитування студентів: 95% з яких вивчали в загальноосвітній школі операційну систему Windows та офісний пакет Microsoft Office, 92% не мають досвіду програмування, 3% мають досвід створення власних програмних проєктів під операційну систему Windows, 5% досвід створення сайтів за допомогою конструкторів сайтів та систем керування контентом.

Проведений аналіз наявних навчальних матеріалів із дисципліни «Вступ до спеціальності» (програміст) дає змогу стверджувати, що питання, які стосуються особливостей професійної діяльності програміста, його службової кар'єри, правового захисту продуктів діяльності, практично не висвітлювались в академічній літературі. Це зумовило необхідність переструктурування змісту навчальної дисципліни.

Таким чином, вивчення дисципліни «Вступ до спеціальності» (програміст) потрібно починати з підвищення рівня мотивації до показників, з якими студенти отримують змогу до самонавчання і самопланування з поступовим отриманням фаху інженера-програміста. Прикладом такої мотивації може бути аналіз студентами під час вивчення дисципліни баз даних вакансій із метою виявлення затребуваності програмістів на ринку праці України та за кордоном, аналіз їхньої заробітної платні та компетенцій, якими вони повинні володіти на даний час. Аналіз рекомендується проводити індивідуально кожним студентом під час перших практичних занять із використанням мережі Інтернет. Це надасть змогу не тільки визначити рівень затребуваності програмістів, але й визначити, яким саме програмістом за спеціалізацією хоче стати той чи інший студент.

Згідно з навчальним планом для студентів напряму підготовки «Інформатика» освітнього рівня «бакалавр» дисципліна «Вступ до спеціальності» є нормативною дисципліною. Загальною метою дисципліни є **надання майбутнім спеціалістам у галузі програмування уявлення про їхню майбутню спеціальність, структуру навчальної програми і місце кожної з дисциплін, що вивчаються, у загальній схемі навчання.**

Мета курсу реалізується у конкретних освітніх та навчальних завданнях:

– сформувати орієнтовний професійний світогляд майбутнього фахівця з інформаційних технологій та цілісне уявлення про його сутність та роль в сучасному суспільстві;

– ознайомити студентів з основними засобами і методами сучасних інформаційних технологій, їх теоретичною і технічною базою, можливими напрямками використання;

– сформувати у студентів знання, вміння та навички, необхідні для подальшого самостійного вдосконалення та самоосвіти з метою ефективного використання засобів розробки програмного забезпечення у своїй майбутній діяльності;

– організувати творчу активність студентів у процесі створення власних програмних розробок;

– надати можливість кожному студенту реалізувати власну навчальну траєкторію шляхом диференціації навчальних завдань та їх творчого характеру;

– сформувати у студентів основи інформаційної культури майбутнього програміста.

Важливість наведених вище навчальних та освітніх завдань вимагає пріоритетного статусу даної дисципліни серед навчальних програм вищих навчальних закладів, в яких відбувається підготовка майбутніх інженерів-програмістів. На факультеті інформатики, математики та економіки МДПУ імені Богдана Хмельницького відведено лише 30 годин на аудиторну роботу, з них 16 годин - на лекційні і 14 годин - на практичні заняття, при значній кількості на самостійну роботу (42 год.), що вимагає чіткої постановки завдань для самостійного опрацювання та визначення форм і термінів їх перевірки. За результатами опитування студентів під час вивчення дисципліни було зроблено висновок про необхідність розширення змісту дисципліни та збільшення аудиторних годин у навчальному плані.

Сучасний рівень використання інформаційно-комунікаційних технологій у навчальному процесі дозволяє викладачу вести електронні конспекти занять (наприклад використовуючи LMS Moodle та електронний підручник), постійно оновлювати їхній зміст

без зайвих витрат на перевидання. Студенти, які з тих чи інших причин пропустили заняття або хочуть звернутися до навчального матеріалу повторно, можуть самостійно користуватися електронними конспектами в зручний для них час. При виконанні навчальних завдань студенти можуть дистанційно консультуватися з приводу різноманітних питань один з одним та з викладачем. При вивченні дисципліни студенти активно використовують електронний підручник із дисципліни, де вони отримують всі необхідні матеріали та проходять тестування за темами, підсумкове тестування студенти проходять у «Електронному журналі», який є модулем інформаційно-аналітичної системи «Університет», таким чином адміністрація університету має можливість проводити постійний моніторинг навчальних досягнень студентів, порівнюючи дані з дисципліни із загальними показниками успішності.

Зміст лекційного блоку знайомить студентів із предметом діяльності програміста, розкриває особливості його професійної діяльності. Тут подається класифікація програмного забезпечення, розглядаються основи теорії програмування, види і типи сучасних мов програмування; подано коротку характеристику деяких систем управління базами даних. Викладаються основні етапи життєвого циклу створення та супроводження програмних продуктів, організації колективів програмістів, відомості про екстремальне програмування, описано види тестування програмного забезпечення та надано поради щодо складання технічного завдання. Висвітлено питання правового захисту комп'ютерної інформації, стандартизації та ліцензування програмних продуктів. Теоретична частина курсу також містить деякі практичні рекомендації з пошуку роботи.

Матеріал структурований у чотири основних модулі: програмування; створення програмного продукту; стандартизація, сертифікація і ліцензування програмного продукту; методологія працевлаштування та професійні захворювання.

На основі аналізу науково-методичної літератури нами було сформульовано вимоги до практичних завдань для студентів, якими ми керувалися у процесі розробки курсу:

1. Використання у процесі навчання інформаційно-комунікаційних технологій, з одного боку, сприяє систематизації знань студента у тій чи іншій галузі, з іншого боку, при виконанні більшості завдань знижує його творчу активність. Тому для компенсації такого впливу ІКТ необхідно використовувати методику створення творчих робіт студентом.

2. Завдання мають бути спрямовані на формування навичок самоосвіти, яка є необхідною умовою подальшого професійного розвитку.

3. Завдання для студентів мають підвищувати внутрішню мотивацію до навчання.

4. Бути професійно зорієнтованими.

5. Формувати усвідомлення існуючих тенденцій розвитку технологій програмування.

6. Надавати цілісне розуміння всіх етапів розробки програмного забезпечення на різних його етапах.

Практичний блок містить вісім практичних завдань. Серед них є такі: навчитися знаходити і відбирати необхідну інформацію за допомогою різноманітних пошукових систем, оформити технічне завдання в установленому порядку та згідно з вимогами ДСТУ/ГОСТ, визначати цілі створення програми, вимоги до програми і основні вихідні дані, необхідні для її розробки, підготувати документи на отримання авторського права на програмний продукт, створити власне портфоліо, написати резюме, пройти декілька сертифікаційних іспитів із метою отримання сертифікату тощо.

Практичні роботи мають типову структуру: назва, мета, обладнання, короткі теоретичні відомості, покрокова проілюстрована інструкція, запитання для самоперевірки засвоєного матеріалу. На допомогу студенту також наведені гіперпосилання на довідкові веб-ресурси.

Під час практичних занять відбувається залучення студентів до розробки тестового програмного продукту, коли кожний з них під час практичних занять отримує свою функціональну роль розробника програмного продукту (згідно моделі MSF), що спонукає їх до подальшого обміну досвідом та доробками з іншими студентами. Кожний із студентів

(груп студентів) відповідно до свого завдання публічно презентує результати своєї роботи, що дозволяє оцінити її іншим учасниками проекту та сформувати командний спосіб роботи.

Одним із різновидів завдань при вивченні цієї дисципліни також є розробка «Діаграми Ганта», спрямованої на вивчення самої дисципліни, таким чином студент при створенні діаграми розплановує етапи та кроки з освоєння навчального матеріалу з прив'язкою до календарних термінів згідно з обсягом та індивідуальною швидкістю навчання.

Завдання для самостійної роботи не диференційовані і не вимагають високого рівня володіння технологіями програмування, спрямовані на розуміння місця програміста на ринку праці та усвідомлення його ролі у підвищенні ролі ІТ-галузі в країні. На самостійну роботу відведена значна кількість годин (42 години), що вимагає її чіткої організації (постановки завдань, форм контролю, термінів виконання). Саме тому використання електронного підручника для забезпечення самостійної роботи дозволило студентам максимально ефективно засвоювати цей матеріал поза навчальним процесом.

Додатково для забезпечення навчального процесу з дисципліни було розроблено навчальний посібник та електронний підручник із дисципліни.

Навчальний посібник "Вступ до спеціальності програміста" (автори: Осадчий В.В., Осадча К.П., Сердюк І.М.) об'єднує фундаментальні основи академічного викладення матеріалу та практичний досвід підготовки програмістів для ІТ індустрії [5].

Посібник включає до свого складу сім розділів. Місце програмування в суспільстві показано в матеріалах першого розділу.

У другому розділі подається класифікація програмного забезпечення.

Основи теорії програмування, види і типи сучасних мов програмування, підходи до програмування, питання кросплатформності розглядаються у третьому розділі. Тут також подано коротку характеристику деяких систем управління базами даних та рейтинг мов програмування.

У четвертому розділі надана інформація про етапи створення програмного продукту, моделі MSF, описані види тестування програмного забезпечення, надано поради щодо складання технічного завдання та написання "красивого програмного коду".

У п'ятому розділі висвітлено питання стандартизації, сертифікації та ліцензування програмного продукту.

Також надається інформація щодо професійних захворювань програмістів.

Матеріали та поради сьомого розділу допоможуть написати гідне резюме, оформити портфоліо, знайти центр отримання сертифікату з підтвердженням кваліфікації ІТ фахівця та підвищення кваліфікації.

Кожний розділ закінчується списком запитань для самоконтролю.

Посібник доповнено словником термінів, списком скорочень та додатками з переліком національних стандартів України щодо розробки програмного продукту і системної документації, прикладами оформлення ліцензійної угоди, заявки на проведення добровільної сертифікації продукції, резюме програміста, технічного завдання.

Також на допомогу студентам у самостійному засвоєнні матеріалу розроблено та впроваджено електронний підручник "Вступ до спеціальності програміста".

На головній веб-сторінці представлені анотація до курсу та його чітка структура у вигляді блоків із гіперпосиланнями. В електронний підручник входять матеріали: теоретичний, практичний та для самостійної роботи. Також в нього входять чотири блоки тестового контролю, поділеного за модулями, додаткові відеоматеріали та більше 62 найменувань рекомендованої літератури в електронному вигляді, що надає можливість ознайомитися з провідними джерелами з цієї тематики (матеріали наведені з навчальною метою для ознайомлення студентів). Такий розподіл матеріалів дозволяє надати студентам велику кількість матеріалів та мотивувати їх до його самостійного засвоєння.

На допомогу студентам автори розробили додаткові розділи "Відеоматеріали", "Глосарій", "Література". Розділ "Тестування" курсу допомагає викладачу організувати перевірку засвоєного матеріалу. Поточне тестування проводиться в електронному

підручнику та є цілодобово доступним, що надає змогу студентам за результатами засвоєння матеріалу перевірити власні знання. Тестовий матеріал поділений на чотири блоки та містить різну кількість запитань, різної складності, відповідно до змісту матеріалу, який вивчався.

Підсумковий тестовий контроль проводиться в «Електронному журналі», та містить 2 тести, кожний по 50 запитань. Для кожного з тестувань використовується випадкова вибірка із загальної бази запитань дисципліни, виходячи з пропорційного відношення запитань, які задаються до загальної кількості запитань - 1/6, з відповідною корекцією запитань для кожного студента, згідно з його початковими показниками при вивченні дисципліни. На підсумкові тестування відведено 50 хвилин. Мета підсумкового тестування - перевірка засвоєння основних положень навчального матеріалу (теоретичний, практичний та для самостійного засвоєння). Кожен студент має тільки одну спробу скласти тест.

Метою вихідного опитування є визначення рівня засвоєння навчального матеріалу та визначення напрямків удосконалення навчального курсу.

Даний навчальний курс мотивує студентів до підвищення якості професійної підготовки та формування цілей та шляху перспективного зростання як майбутніх програмістів, виховує навички створювати коректний програмний код та повагу до авторського права.

Таким чином, розроблений нами навчальний курс «Вступ до спеціальності» в повній мірі відповідає сучасному світовому рівню організації навчального процесу у вишах. Курс можна «тиражувати» в інших навчальних закладах, разом із тим, електронна форма курсу є відкритою для модифікації та вдосконалення.

Апробація курсу. Згідно з планом-регламентом роботи МДПУ імені Богдана Хмельницького на 2010-2011, 2011-2012, 2012–2013, 2013-2014 навчальні роки дана дисципліна викладалась в період з 1 вересня по 31 грудня (2 модулі) студентам I курсу факультету інформатики і математики. Загалом протягом 2010-2013 років дану дисципліну вивчали 57 студентів.

Результати фінального опитування студентів свідчать про осмислення ними себе як майбутнього програміста, розуміння основних етапів своєї підготовки та професійного становлення як майбутнього фахівця. Студенти оцінили матеріали курсу «Вступ до спеціальності» (інженер-програміст) у середньому на 9,35 балів з 10 можливих, новизну лекційного матеріалу - у 8,71 балів, та новизну практичних завдань – у 9,04 балів за десятибальною шкалою.

Таким чином, апробація підтвердила позитивний результат нашої роботи.

Щодо подальшого вдосконалення змістовної складової курсу можна спиратися на побажання студентів, висловлені ними в ході заключного опитування. Узагальнюючи їхні відповіді, до розширеного варіанту курсу - за умови збільшення годин - доцільно включити ознайомлення з методологіями керування програмними проектами: Agile, Scrum, RUP тощо. Також серед відповідей студентів часто зустрічається бажання глибше вивчити різноманітні інструменти керування проектами (Trac, Readmine, Mantis, Git) [6] та методику розрахунку економічної ефективності майбутніх програмних проектів.

Висновки. За результатами розробки та апробації навчального курсу «Вступ до спеціальності» (інженер-програміст) виявлено та доведено, що під час навчання підвищується якість професійної підготовки студентів, відбувається формування цілей та шляхів їх перспективного зростання як майбутніх програмістів. Можна стверджувати, що внесені зміни до структури та змісту навчальної дисципліни, а також створене навчально-методичне забезпечення (навчальний посібник та електронний підручник) в повній мірі відповідає сучасному рівню організації навчального процесу у вишах та дозволяє сформувати в майбутніх програмістів розуміння процесу оволодіння необхідними професійними компетентностями.

Список використаних джерел

1. Поурочне календарне планування з інформатики для 11 класу на 2010/2011 / Освіта України [Електронний ресурс]. – Режим доступу: <http://fij.com.ua/kalendarnie-planuvannya/1868-informatika-kalendarni-plani-dlya-11-klasu.html>
2. Навчальна програма поглибленого вивчення інформатики для учнів 10-11 класів загальноосвітніх навчальних закладів / Програми інформатика [Електронний ресурс]. – Режим доступу: informatic.sumy.ua/downloads/zbirnuk_program_inf/inf_pogl.docx
3. Освітньо-кваліфікаційна характеристика бакалавра за напрямом підготовки 040302 «Інформатика». Стандарт вищої освіти. – К.: Міністерство освіти і науки України, 2010. – 32 с.
4. Освітньо-кваліфікаційна характеристика бакалавра за напрямом підготовки 050103 «Програмна інженерія». Стандарт вищої освіти. – К.: Міністерство освіти і науки України, 2008. – 20 с.
5. Осадчий В. В. Вступ до спеціальності програміста : навч. посіб. / В. В. Осадчий, К. П. Осадча, І. М. Сердюк. – Мелітополь: Вид. будинок ММД, 2011. – 296 с.
6. Осадчий В. В. Основи розробки веб-додатків: навч. посіб. / В. В. Осадчий, В.С. Круглик. – Мелітополь: Вид. будинок ММД, 2012. – 540 с.
7. Інформатика. Навчальна програма для учнів 10-11 класів загальноосвітніх навчальних закладів. Рівень стандарту / Програми інформатика [Електронний ресурс]. – Режим доступу: informatic.sumy.ua/downloads/zbirnuk_program_inf/inf_st.docx
8. Інформатика. Навчальна програма для 10-11 класів загальноосвітніх навчальних закладів. Академічний рівень / Програми інформатика [Електронний ресурс]. – Режим доступу: informatic.sumy.ua/downloads/zbirnuk_program_inf/inf_ak.docx

Осадчий В.В., Осадча К.П.

Досвід розробки та апробації навчального курсу «Вступ до спеціальності» (інженер-програміст)

У статті висвітлено вагоме значення вивчення навчальної дисципліни «Вступ до спеціальності» (програміст). Проведений аналіз навчальних планів загальноосвітніх шкіл з інформатики; проаналізовано навчальні матеріали з дисципліни та вимоги стандартів щодо професійної підготовки програмістів. Відповідно до змісту представленого курсу визначена мета та навчальні завдання; охарактеризовано структуру розроблених компонентів навчального курсу «Вступ до спеціальності» (програміст): друкований посібник, електронний підручник та підсумковий тестовий контроль в «Електронному журналі». Акцентовано увагу на процесі підготовки майбутніх програмістів відповідно до запитів ринку праці, для чого пропонується вивчення технологій розробки програмного забезпечення; сертифікації знань; створення портфоліо, резюме та технічного завдання тощо. Також у статті подана апробація курсу та результати фінального опитування студентів.

Ключові слова: інженер-програміст, навчальний курс, електронний підручник, тестовий контроль.

Осадчий В.В., Осадчая К.П.

Опыт разработки и апробации учебного курса « Введение в специальность » (инженер - программист)

В статье рассмотрено ведущее значение изучения учебной дисциплины «Введение в специальность» (программист). Проведен анализ учебных планов общеобразовательных школ по информатике; проанализированы учебные материалы по дисциплине и требования стандартов к профессиональной подготовке программистов. Согласно содержанию представленного курса определена цель и учебные задачи; охарактеризовано структуру разработанных компонентов учебного курса «Введение в специальность» (программист): учебник, электронный учебник и итоговый тестовый контроль в «Электронном журнале». Акцентируется внимание на процессе подготовки будущих программистов в соответствии с запросами рынка труда, для чего предлагается изучение технологий разработки

программного забезпечення; сертифікації знань; створення портфоліо, резюме і технічного завдання і т.п.. Також в статті представлена апробація курсу і результати фінального опроса студентів.

Ключевые слова: инженер - программист, учебный курс, электронный учебник, тестовый контроль.

V. Osadchyi, K. Osadcha

Experience in Developing and Testing the Course "Introduction to the Profession" (Software Engineer)

The article discusses the importance of studying the master discipline «Introduction to the profession» (Software Engineer). The paper showed the analysis of the curricula of secondary schools in computer science; analyzed educational materials on the discipline and the standards to training programmers. According to the course content presented definite purpose and learning objectives were considered; it was characterized the structure of developed components of the course «Introduction to the profession» (Software Engineer) textbook, electronic textbook and final test in control «Electronic journals». The attention is focused on the preparation of future programmers in accordance with the needs of the labor market, which is proposed to study the software technologies; on certification of knowledge; on creation a portfolio, resume and technical specifications etc. The article also presented the results of testing of the course and the final students` survey.

Keywords: Software Engineer, curriculum, electronic textbooks, test control.

Стаття надійшла до редакції 21.03.2014р