

УДК 004.4+378.147

Л.В. Гришко

ПРО ФОРМУВАННЯ ПРОФЕСІЙНИХ ЯКОСТЕЙ МАЙБУТНЬОГО ПРОГРАМІСТА В ПРОЦЕСІ НАВЧАННЯ ОСНОВ ПРОГРАМУВАННЯ

У роботі розглядаються ключові компетенції, притаманні інженерам-програмістам. Також визначені компетенції (знання, уміння, навички), які можуть бути сформовані у студентів у процесі навчання основ програмування.

Ключові слова: ринок праці, програміст, компетенції, студент, навчальний процес, розвиток особистості.

Сьогодні конкурентноздатність фахівця на сучасному ринку праці багато в чому залежить від його здатності опановувати нові технології, адаптуватися до інших умов праці. Тому провідна ідея системи освіти виражена в спробі зв'язати результуючу складову освіти з її планованими результатами. Впровадження компетентнісного підходу в процес навчання надає можливість реалізації цієї ідеї.

У сучасному уявленні компетентнісного підходу маються на увазі ідеї загального й особистісного розвитку, сформульовані у контексті психолого-педагогічних концепцій розвиваючої і особистісно-орієнтованої освіти, де зміст освіти визначається чотирьохкомпонентною моделлю: знання, уміння, досвід творчої діяльності і досвід ціннісного відношення [10].

Усередині компетентнісного підходу виділяють два базових поняття: компетенція і компетентність, при цьому перше з них „включає сукупність взаємозалежних якостей особистості, що задаються стосовно визначеного кола предметів і процесів”, а друге співвідноситься з „володінням людиною відповідною компетенцією, що включає його особистісне відношення до неї і предмету діяльності” [12]. У цьому ж контексті функціонує і поняття „освітньої компетенції”, яка розуміється як „сукупність значеннєвих орієнтацій, знань, умінь, навичок і досвіду діяльності учня стосовно визначеного кола об'єктів реальної дійсності, необхідних для здійснення особистісної і соціально-значимої продуктивної діяльності”. Освітні компетенції диференціюються за рівнями, як і зміст освіти [10]:

- ключові (реалізовані на метапредметному, загальному для всіх предметів змісті);
- загальнопредметні (реалізовані на змісті, що є інтегративним для сукупності предметів, освітньої галузі);
- предметні (формовані в рамках окремих предметів).

Існують різноманітні підходи до формулювання ключових компетенцій і їхніх класифікацій. Наприклад, російські класифікації включають загальнокультурну, навчально-пізнавальну, інформаційну, комунікативну, соціально-трудову компетенції і компетенцію особистісного самовдосконалення. У системі сертифікації компетенцій, що розроблена OCR (екзаменаційною комісією Оксфорда Кембриджу), виділяються ключові компетенції декількох

рівнів, що, на думку розробників, необхідно враховувати при підготовці компетентного висококваліфікованого фахівця незалежно від професійної спрямованості: 1 – комунікація; 2 – операції з числами; 3 – інформаційні технології; 4 – робота з людьми; 5 – удосконалення здібностей до навчання і підвищення результативності; 6 – вирішення проблем; 7 – розвиток особистісних компетенцій [3].

Питанням професійних якостей програмістів і аналізом специфіки задач у галузі програмування у різний час займалися психологи і педагоги Ф. Брукс [1], Г. Вейнберг, Н. Вірт [2], Е. Дейкстра [5], С. Макконнелл [7], М.Л. Смульсон [11], Б. Шнейдерман [13] та ін. Дослідники визначають, здібності і особливості мислення, які є притаманними програмістові (табл. 1).

Таблиця 1

Якості, які властиві програмістові, що пов'язані безпосередньо із розробкою програмного продукту

	Перелік якостей
За Е.Дейкстрою [4]:	<ul style="list-style-type: none"> – <i>здатність визначити архітектуру програми</i>, тобто розбивати складну задачу на елементарні складові і задати варіанти їх комбінування; – <i>уміння бачити задачу одночасно на різних рівнях деталізації</i>, тобто програміст повинен вільно переходити від опису задачі у загальних поняттях до суті нижчого рівня, що стоїть за цими поняттями; – <i>уміння уявляти собі процес, який проектується, у динаміці</i>, оскільки, дані, що оброблюються, у деякий момент часу можуть мати одні значення і взаємозв'язки, а у наступний момент деякі з них можуть змінитися; – <i>уміння бачити далі за одну програму, що розробляється в даний момент</i>, тобто вміти враховувати широке оточення даної частини задачі і можливість її включення до деякої загальної системи; – <i>уміння узагальнювати типові ситуації</i>, тобто необхідно вміти знаходити в програмі ідейно однорідні ділянки; – <i>уміння застосовувати і комбінувати добре відомі прийоми програмування і типові алгоритми</i>, тобто більшість нових ідей повинна знаходитися в тісній взаємодії з вже відомими ідеями і методами; – <i>уміння модифікувати програму</i>, оскільки постановка задачі в процесі розв'язування уточнюється і розроблена програма, яка вже написана, може не розв'язувати правильно поставлену задачу.

Продовження Табл. 1

	Перелік якостей
За Б. Шнейдерманом [13]:	<ul style="list-style-type: none"> – здатність розуміти програми; у програмуванні розуміння має нижній рівень (розуміння кожного рядка коду), середній рівень (розуміння структури алгоритму і даних) і вищий рівень (розуміння загального призначення програми); – <i>здатність налагоджувати програми</i>, тобто знаходити помилки в програмі. Така робота містить елементи розв’язування головоломок. Складність такої роботи обумовлена психологічними чинниками – посиленням занепокоєнням і небажанням допускати помилки; – <i>здатність модифікувати програми</i>: модифікація програм є досить поширеною задачею програмування. Успішність у вирішенні задачі модифікації можна вважати одним з критеріїв розуміння програм; – <i>уміння запам’ятовувати і відтворювати текст програм</i>: здатність відтворення правильної програми „по пам’яті” знаходиться у прямій залежності від досвіду програмування.
За М.Л. Смульсон [11]:	<ul style="list-style-type: none"> – <i>гнучкість і стратегічність мислення</i>; – <i>творчі властивості мислення</i>; – <i>уважність</i>, що виявляється у вмінні припускати якнайменше помилок; – <i>логічний характер мислення</i>.

Формування необхідних професійних якостей пов’язано з вихованням деяких загальнолюдських рис характеру у студентів, майбутніх спеціалістів у галузі інформаційних технологій. Б. Шнейдерман відзначає, що „Особливості особистості відіграють критичну роль у визначенні як взаємодії між програмістами, так і робочого стилю окремих програмістів” [13] (табл. 2.)

Таблиця 2

Психологічні і загальнолюдські риси, які повинні бути притаманні програмістові

	Перелік якостей
За Е. Дейкстрою [4]:	<ul style="list-style-type: none"> – <i>наявність комплексного мислення</i> - уміння наперед визначати етапи, які потрібно пройти, щоб вирішити те або інше питання; – <i>здатність аналізувати власні помилки</i> (ця риса вказує на вимогливість програміста до себе з одного боку і застосування стилю програмування, що зменшує кількість помилок, з іншого боку);

Продовження Табл. 2

	Перелік якостей
	<ul style="list-style-type: none"> – <i>культура власної праці</i> - уміння забезпечити себе необхідним інструментарієм для роботи; – <i>уміння працювати в колективі</i>, оскільки практично будь-яка серйозна розробка носить колективний характер, при цьому успіх всієї роботи залежить від взаєморозуміння, розподілу функцій і взаємостосунків у колективі; – <i>уміння працювати з користувачем</i>: професійний програміст повинен розуміти потреби користувача, уміти оцінити зручність конкретних форм інтерфейсу, мати змогу і здатність привчити користувача до нових засобів і систем; програміст повинен володіти психологічними знаннями, що допомагають йому у спілкуванні з користувачем; – <i>володіння інтелектуальними засобами</i>, які застосовуються для розуміння програми; – <i>дотримання правил загальнолюдської етики</i>; – <i>здатність чітко бачити дійсні труднощі і відкидати все, що не відноситься до справи</i>; – <i>здатність виявляти всі випадки, де можна застосувати теорію</i>, самостійно зважитися на її застосування або звернутися за порадою до більш досвідченого програміста; – <i>здатність при невдачі подолати самолюбство і знайти інший підхід</i>.
За Б.Шнейдерманом[13]:	<ul style="list-style-type: none"> – <i>наполегливість/насивність</i>: наполеглива людина володіє необхідною для виконання роботи ініціативою; – <i>інтровертність/екстровертність</i>: при колективному співробітництві й при взаємодії з користувачем сприятливим є дружній стиль відносин; – <i>внутрішня/зовнішня керованість</i>: особистості з вираженою внутрішньою керованістю намагаються підкорити собі обставини; – <i>висока/низька збудливість</i>: помірний рівень збудливості сприяє підвищенню продуктивності; – <i>висока/низька мотивація</i>: особистості з високим рівнем мотивації здатні виконувати дуже складні завдання;

Продовження Табл. 2

	<ul style="list-style-type: none"> – <i>висока/низька терпимість до невизначеності</i>: розробники повинні вміти працювати в умовах, коли: <ul style="list-style-type: none"> – чітко встановлене невелике число фактів або компонентів; – потрібно приймати рішення при обмежених вхідних даних; – для прийняття чергового рішення потрібно мати схильність до деякого ризику; – <i>уміння бути точним</i>: на завершальних етапах розв’язування завдання необхідні особлива увага до подробиць і готовність перевірити й урахувати кожну деталь; – <i>скромність</i>. гарний програміст не повинен занадто самовпевнено ставитися до якості своїх програм; – <i>здатність переносити стрес</i>. У випадках відставання від планових строків потрібна здатність добре працювати у стресовій ситуації.
За М.Л.Смультсон [8 - 190]:	<ul style="list-style-type: none"> – висока працездатність і ретельність у праці; – оперативність мислення; – вміння приймати рішення в умовах обмеженого часу; – вміння створювати собі робоче місце, яке сприяє підвищенню – продуктивності праці

У роботі [9] поняття «професіоналізм» по відношенню до програміста визначається як інтегральна особистісна характеристика людини, яка:

- оволоділа нормами професійної діяльності і спілкування та здійснює їх на високому рівні, добиваючись професійної майстерності в галузі програмування;
- слідує професійній ціннісній орієнтації, дотримуючись професійної етики;
- розвиває себе засобами професії;
- прагне зробити творчий внесок у професію, збагативши її досвід;
- прагне і вміє викликати інтерес суспільства до результатів своєї професійної діяльності, сприяє підвищенню ваги і престижу своєї професії в суспільстві, гнучко враховує нові запити суспільства до неї. У цілому, „успішність людини у майбутньому професійному й соціальному житті визначається рівнем розвитку ключових компетентностей” [8].

Під компетентністним підходом у навчанні основ програмування ми розуміємо підхід, що акцентує увагу на результатах навчання. Як результат будемо розглядати не суму засвоєних знань, а формування у студентів здатності використовувати отримані знання з основ програмування в процесі навчання професійно-орієнтованих дисциплін й у наступній професійній діяльності.

На нашу думку, компетентності з основ програмування – це наявність у студента здатностей:

- загальних:
 - використовувати в практичній діяльності засвоєні знання, уміння та навички з основ програмування у процесі вивчення дисциплін з комп'ютерних наук, у навчальному процесі в цілому та у наступній професійній діяльності
 - організовувати й планувати свою, зокрема навчальну, діяльність;
 - знаходити й аналізувати відомості з різних джерел;
 - формалізувати здобути знання, накопичувати їх;
 - працювати в колективі;
 - використовувати знання іноземних мов у професійної діяльності.
- спеціальних:
 - складати алгоритми поставлених задач и писати програми;
 - застосовувати й комбінувати типові алгоритми й відомі методи програмування;
 - налагоджувати й тестувати навчальні програми в середовищі розробки мови програмування;
 - розуміти галузі застосування здобутих знань з основ програмування в інших професійно-орієнтованих і спеціальних дисциплінах та в сфері майбутньої професійної діяльності;
 - проектувати і розробляти невеличкі програмні продукти індивідуально або групою з 2-3 осіб;
 - вивчати нові мови програмування.

Погоджуючись із наведеними вище вимогами до професійних якостей програміста, ми вважаємо, що до числа особливо важливих характеристик програміста варто віднести *здатність до навчання протягом усього періоду професійної діяльності*.

На підставі власного досвіду також вважаємо, що перераховані якості можна і треба сформулювати у майбутніх інженерів-програмістів вже в курсі з основ програмування. Для забезпечення цього, крім визначення цілей і змісту навчання основ програмування, доцільно:

- дотримуватися принципів систематичності і послідовності навчання, свідомості, активності і самостійності студентів у навчанні при керівній ролі викладача;
- здійснювати розподіл змісту курсу з основ програмування на дві частини – інваріантну, однакову для всіх напрямів підготовки в галузі інформатики та обчислювальної техніки, та варіативну, зміст якої залежить від конкретної спеціальності;
- впроваджувати пропедевтичний курс з метою забезпечення у студентів необхідного стартового рівня знань, умінь і навичок з алгоритмізації і програмування з використанням навчальної мови програмування Pascal,;
- з міркувань перспективності застосування програмування у професійній діяльності в курсі з основ програмування треба вчити студентів програмувати мовою програмування C;

- у процесі навчання основ програмування змістити акцент з вивчення синтаксичних та семантичних особливостей мов програмування на вивчення методів побудови й аналізу алгоритмів, використання задачного підходу до навчання основ програмування на базі широкого спектру різноманітних видів і типів навчальних завдань з основ програмування різного рівня складності;
- удосконалювати традиційні методи навчання, а також запроваджувати методи інтерактивного і активного навчання, зокрема, створення студентами (групою у 2-3 особи) навчального програмного продукту під керівництвом викладача;
- використовувати засоби навчання (посібники, підручники, методичні розробки, ППЗ, технічні засоби навчання, інформаційно-комунікаційні технології) разом з іншими компонентами методичної системи як єдиний комплекс, що забезпечує вирішення задачі навчання основ програмування;
- впроваджувати індивідуальну і диференційовану форми організації навчання;
- створювати умови для організації самостійної роботи студентів у поза аудиторний час у комп'ютерному класі або поза межами навчального закладу з використанням технологій дистанційного навчання;
- впроваджувати модульно-рейтингову систему контролю і оцінювання знань студентів з метою підвищення і можливості прогнозування результатів навчання.

Впровадження в навчальний процес запропонованих засобів на основі сучасних педагогічних та інформаційно-комунікаційних технологій надає можливість інтенсифікувати процес навчання, суттєво активізувати пізнавальну діяльність студентів, підвищити рівень базових знань студентів з основ програмування, диференціювати та індивідуалізувати процес навчання з огляду на різний рівень здібностей студентів, підвищити рівень підготовленості студентів до вивчення професійно-орієнтованих дисциплін, підвищити рівень їх професійної підготовки [3].

Література:

1. Брукс Ф.П. Как проектируются и создаются программные комплексы / Брукс Ф.П.; [пер. с англ.]. – М. : Наука, 1979. – 151 с.
2. Вирт Н. Алгоритмы + структуры данных = программы / Вирт Н. ; [пер. с англ.]. – М. : Мир, 1985. – 406 с.
3. Гришко Л.В. Методична система навчання основ програмування майбутніх інженерів-програмістів : Дис. канд. пед. наук: 13.00.02 / НПУ ім. Драгоманова. – К., 2009. - 254 с.
4. Дейкстра Э. Заметки по структурному программированию // Дал У., Дейкстра Э., Хоор К. Структурное программирование. – М.: Мир, 1975. – 247 с.
5. Дейкстра Э. Дисциплина программирования: Пер. с англ. – М.: Издательство “Мир”, 1978. – 274 с.
6. Ключевые компетенции 2000. Программа. Уровни 1, 2, 3, 4, 5. KEY SKILLS 87 0005 (OCR. RECOGNISING ACHIEVMENT. – Oxford Cambridge and RSA Examinations).
7. Макконелл Дж. Анализ алгоритмов, Вводный курс. – Москва: Техносфера, 2002. – 304 с.
8. Мозолин В. В. Компетентностный подход в обучении информатике: возможности и проблемы/В. В. Мозолин//Информатика и образование. – 2005. – №2. – С. 115 – 116.
9. Одинцов И.О. Профессиональное программирование. Системный подход. – СПб.: БХВ-Петербург, 2002. – 512 с.

10. Проблемы и перспективы реализации компетентного подхода в образовании : (Проблемы компетентного подхода) [Электронный ресурс] / А.Г. Бермус // Интернет-журнал «Эйдос» – Научные исследования. – Режим доступа до журн. : <http://www.auditorium.ru>.
11. Смутьсон М.Л. Психология розвитку інтелекту: Монографія. – К., 2001. – 276 с.
12. Хуторской А.В. Ключевые компетенции и образовательные стандарты // Интернет-журнал "Эйдос". – 2002. – 23 апреля. <http://www.eidos.ru/journal/2002/0423.htm>.
13. Шнейдерман Б. Психология программирования: Человеческие факторы в вычислительных и информационных системах. Пер. с англ.– М.: Радио и связь, 1984. – 304 с.

Прийнято до друку 31.10.2011

Аннотация

Гришко Л.В. О формировании профессиональных качеств будущего программиста в процессе обучения основам программирования.

В работе рассматриваются ключевые компетенции, которые присущи инженерам-программистам. Также определены компетенции (знания, умения и навыки), которые могут быть сформированы у студентов в процессе обучения основам программирования.

Ключевые слова: рынок труда, программист, компетенции, студент, учебный процесс, развитие личности.

Summary

Grishko L.V. On the formation of professional skills of the future programmer in learning the basics of programming.

In this work the core competencies inherent programming engineer. Also identified competencies (knowledge, abilities, skills) that might be formed by the students in learning programming basics.

Keywords: labor market, the programmer, the competence of the student, the learning process, the development of lichnosti.