

системою : монографія / Спірін О. М. ; [наук. ред. акад. М. І. Жалдака]. – Житомир : Вид-во ЖДУ ім. І. Франка, 2007. – 300 с.

6. Педагогіка: методи навчання [Електронний ресурс] // Навчальні матеріали онлайн. – Режим доступу : [http://pidruchniki.com/19670318/pedagogika/metodi\\_navchannya](http://pidruchniki.com/19670318/pedagogika/metodi_navchannya)

7. Шеремета П. М. Кейс-метод: з досвіду викладання в українській бізнес-школі / П. М. Шеремета, Л. Г. Каніщенко ; за ред. О. І. Сидоренка. – 2-ге вид. – К. : Центр інновацій та розвитку, 1999. – 80 с.

**Конюхов С.,  
Храпач О.**

*Мелітопольський державний педагогічний університет  
імені Богдана Хмельницького*

## ОГЛЯД ОСНОВНИХ ПОНЯТЬ ІНЖЕНЕРІЇ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Постановка проблеми в загальному вигляді. Інформаційні технології обслуговують всі сфери функціонування суспільства, тому до якості програмного забезпечення (ПЗ) висувається комплекс вимог, дотримання яких необхідно відстежувати на всіх етапах життєвого циклу програмних систем (ПС). Якість програмного продукту можливо гарантувати за умови ефективної організації процесу розробки із застосуванням методів оптимізації і всебічного тестування його параметрів.

Підприємства галузі інформаційних технологій мають потребу у кваліфікованих фахівцях з розробки і тестування програмного забезпечення. Це пояснюється важливим значенням, яке процеси верифікації, тестування і оптимізації відіграють у життєвому циклі програмних систем. У зв'язку з цим, одним із завдань підготовки фахівців з інформаційних технологій у вищих навчальних закладах є формування компетентностей з верифікації, тестування і оптимізації програмного забезпечення.

Аналіз останніх досліджень і публікацій. Проблеми верифікації і тестування ПЗ розглядають І. Будз, Л. Гнатів, Т. Говорущенко, О. Дерев'янченко, Ю. Карпов, В. Кукурба, В. Кулямін, К. Лавріщева, Є. Луценко, О. Щербаков, В. Яковина. Питання удосконалення методів оптимізації програмного забезпечення досліджують В. Бельська, Ю. Зілінський, С. Макконнелл, О. Молчанов, А. Юдіна та ін. Різні аспекти викладання у вищих навчальних закладах

дисциплін, пов'язаних з цією проблематикою, розглядають А. Адаманський, М. Безверха, С. Гайсарян, Ф. Ільєсова, К. Кулаков, К. Лавріщева, К. Осадча, В. Осадчий, М. Сидоров, Ю.І. Скорін, Е. Усеїнов, І. Ясенова-Берестюк та інші дослідники.

Виділення невіршених раніше частин загальної проблеми, яким присвячується стаття. Навчальними планами підготовки фахівців з інформаційних технологій у вишах України вивчення названих технологій передбачається у складі різних дисциплін. У зв'язку з цим, необхідно визначити коло питань, які доцільно вивчати у курсі «Методи верифікації та оптимізації програм», призначеному для студентів спеціальності «Інформатика» МДПУ імені Богдана Хмельницького.

Постановка завдання. В статті будуть розглянуті основні поняття інженерії якості програмного забезпечення з метою подальшого визначення змісту дисципліни «Методи верифікації та оптимізації програм».

Виклад основного матеріалу дослідження.

Наведемо визначення, необхідні для розуміння понять «компетентність» і «компетенція»:

1. «компетентність – динамічна комбінація знань, вмінь і практичних навичок, способів мислення, професійних, світоглядних і громадянських якостей, морально-етичних цінностей, яка визначає здатність особи успішно здійснювати професійну та подальшу навчальну діяльність і є результатом навчання на певному рівні вищої освіти» [2].

2. «Компетенція включає знання й розуміння (теоретичне знання академічної області, здатність знати й розуміти), знання як діяти (практичне й оперативне застосування знань до конкретних ситуацій), знання як бути (цінності як невід'ємна частина способу сприйняття й життя з іншими в соціальному контексті)» [1, с. 13].

Компетенції в галузі тестування, верифікації і оптимізації програм включені до діючих галузевих стандартів вищої освіти за освітнім ступенем „бакалавр” з напрямів, за якими здійснюється підготовка фахівців з інформаційних технологій у вишах України.

Так, наприклад, «Галузевим стандартом вищої освіти України з напрямку підготовки 040302 «Інформатика» передбачено, у бакалаврів цього напрямку підготовки мають бути сформовані такі компетенції [1, с. 20-21]:

1. Загально-професійні компетенції (КЗП):

а) знання вимог чинних державних та міжнародних стандартів, методів і засобів проектування комп'ютеризованих систем, життєвого циклу їх програмного забезпечення (КЗП-4);

б) знання основних методів та підходів щодо організації, планування, керування та контролю роботами з проектування, розроблення, післяпроектного супроводу та експлуатації програмного забезпечення комп'ютеризованих систем (КЗП-7).

2. Спеціалізовано-професійні компетенції (КСП):

а) знання сучасних методів розробки та оптимізації концепцій комп'ютерної реалізації моделей об'єктів і процесів інформатизації (КСП-3);

б) знання методів побудови та верифікації абстрактної архітектури комп'ютеризованої системи та знання апаратних платформ та програмних середовищ, що відповідають побудованій архітектурі (КСП-5);

в) знання методів виявлення, формулювання, специфікації, аналізу та трасування вимог до комп'ютеризованих систем на етапі їх проектування, методів проектування та верифікації абстрактної архітектури комп'ютеризованих систем (КСП-6);

г) знання методів, методик контролю та тестування правильності роботи програмного забезпечення комп'ютеризованих систем (КСП-17).

Життєвий цикл програмного забезпечення – це інтервал часу від моменту виникнення ідеї про розробку або придбання програмної системи для розв'язання певних задач до моменту повного завершення використання останньої її версії [3, с. 6].

«Якість ПЗ – набір властивостей продукту (сервісу або програм), що характеризують його здатність задовольнити встановлені або передбачувані потреби замовника» [5, с. 45].

А. Миронов наводить такі вимоги до якості програмних систем [7, с. 5]:

1. Коректність – відповідність системи своєму призначенню.

2. Безпечність – відсутність неавторизованого витоку інформації під час функціонування системи, а також здатність до швидкого відновлення роботи після збою, що виник в результаті атаки на систему.

3. Стійкість системи у випадку непередбаченої поведінки оточення і використання неправильних вхідних даних.

4. Ефективність використання ресурсів часу і пам'яті. Оптимальність алгоритмів, реалізованих в системі.

5. Здатність системи до адаптації у випадку незначних змін оточення шляхом зміни її налаштувань без зміни її внутрішньої структури.

6. Чітка і зрозуміла документованість внутрішньої структури системи.

7. Переносимість і сумісність, тобто здатність системи ефективно функціонувати на різних платформах і в різних конфігураціях.

Розрізняють три основні види діяльності, спрямовані на досягнення якості програмного забезпечення [4, с. 34]:

1. Інспекція – аналіз і перевірка різних представлень системи і програмного забезпечення (специфікацій, архітектурних схем, діаграм, вихідного коду та ін.), які виконуються на всіх етапах життєвого циклу розробки ПЗ.

2. Верифікація – процес забезпечення правильної реалізації ПЗ, яка відповідає специфікаціям, що виконується протягом всього життєвого циклу. Верифікація відповідає на питання, чи вірно розроблена система.

3. Валідація – процес перевірки відповідності програмного забезпечення функціональним і нефункціональним вимогам і потребам замовника.

Верифікація вирішує такі основні завдання в життєвому циклі програмних систем [3, с. 16-17]:

– виявлення дефектів (помилки, недоробок, неповноти та ін.) різних артефактів розробки ПЗ (вимог, проектних рішень, документації або коду);

– виявлення найбільш критичних і найбільш схильних до помилок частин створюваної або супроводжуваної системи;

– контроль і оцінка усіх аспектів якості ПЗ;

– надання всім зацікавленим особам інформації про поточний стан проекту і характеристики його результатів;

– надання керівництву і розробникам проекту інформації для планування подальших робіт і прийняття рішень про продовження проекту, його припинення або передачу результатів замовнику.

Верифікація програмної системи складається з таких етапів [7, с. 7]:

1. Побудова математичної моделі аналізованої системи.

2. Подання властивостей, які перевіряються, у вигляді формального тексту – специфікації.

3. Побудова формального доказу наявності або відсутності у системи властивості, яка перевіряється.

Детальну оцінку надійності програми надає тестування. Розробка стратегії тестування, яка враховує вимоги до системи, її архітектуру і проект, є частиною загальної системи контролю якості [6, с. 459].

«Тестування – це метод виявлення помилок у ПС шляхом виконання вихідного коду на тестових даних, збирання робочих характеристик у динаміці виконання в конкретному операційному середовищі, виявлення різних помилок, дефектів, відмовлень і збоїв, викликаних нерегулярними, аномальними ситуаціями або аварійним припиненням роботи системи.» [5, с. 165].

Методи тестування програмних систем можна класифікувати за об'єктами тестування [9, с. 170-171]:

1. Модульне тестування: перевірка функціонування окремих елементів системи.

2. Інтеграційне тестування: перевірка взаємодії між програмними компонентами – модулями.

3. Системне тестування. Охоплює повністю всю систему і, як правило, фокусується на нефункціональних вимогах до безпеки, продуктивності, точності, надійності та ін. На цьому рівні також відбувається тестування інтерфейсів до зовнішніх додатків, апаратного забезпечення, операційного середовища та ін.

Для автоматичного проведення тестування і підготовки звітів про наявні помилки використовуються різноманітні програми, наприклад: Selenium, Codeception, TestPlan, Apache JMeter та інші.

Ще одним способом підвищення ефективності і якості програм є оптимізація коду. С. Макконелл вказує, що «Оптимізацією коду називають зміну коректного коду, спрямовану на підвищення його ефективності. „Оптимізація” передбачає внесення невеликих змін, які стосуються одного класу, одного методу, а найчастіше – кілька рядків коду» [6, с. 576].

Критеріями ефективності отриманої програми є об'єм пам'яті, необхідний для її роботи (для зберігання всіх даних і коду), і швидкість виконання. Разом із тим, в процесі оптимізації важко задовольнити обидва критерії, тому, як правило, обирають більш

важливий для даного випадку або формулюють комплексний критерій [8, с. 252].

Висновки. Застосування методів верифікації, тестування і оптимізації в процесі розробки є важливим чинником забезпечення якості, надійності і ефективності отриманих програмних продуктів. У зв'язку з цим, необхідно здійснювати підготовку майбутніх фахівців з інформаційних технологій у вищих навчальних закладах до їх використання в професійній діяльності. Для досягнення цієї мети подальші дослідження слід спрямувати на визначення структури і змісту дисципліни «Методи верифікації та оптимізації програм» і розробку інтерактивних засобів підтримки її викладання.

#### СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Галузевий стандарт вищої освіти України. Освітньо-кваліфікаційна характеристика: [освітньо-кваліфікаційний рівень] бакалавр. Галузь знань 0403 «Системні науки та кібернетика». Напрямок підготовки 040302 «Інформатика». Кваліфікація 3121 Фахівець з інформаційних технологій. 3340 Викладач-стажист / Міністерство освіти і науки України. – К., 2010. – 32 с.

2. Закон України «Про вищу освіту»: за станом на 16.01.2016 р.: [Електронний ресурс] / Відомості Верховної Ради (ВВР). – 2014. – № 37-38, ст.2004. – Режим доступу: <http://zakon3.rada.gov.ua/laws/show/1556-18/page>.

3. Кулямин В.В. Методы верификации программного обеспечения: [Електронний ресурс] / В.В. Кулямин. – М.: Институт системного программирования РАН. – 117 с. – Режим доступу: [www.ict.edu.ru/ft/005645/62322e1-st09.pdf](http://www.ict.edu.ru/ft/005645/62322e1-st09.pdf).

4. Лаврищева Е.М. Методы и средства инженерии программного обеспечения. Учебник / Е.М. Лаврищева, В.А. Петрухин. – М.: Московский физико-технический институт (государственный университет), 2006. – 304 с.

5. Лаврищева К.М. Програмна інженерія. Підручник / К.М. Лаврищева. – К.: Видавничий дім «Академперіодика» НАН України, 2008. – 319 с.

6. Макконнелл С. Совершенный код. Мастер-класс / С. Макконнелл. – М.: Издательство «Русская редакция», 2010. – 896 с.

7. Миронов А.М. Верификация программ методом Model Checking: [Електронний ресурс] / А.М. Миронов. – 83 с. – Режим доступу: <http://intsys.msu.ru/staff/mironov/modelchk.pdf>.

8. Молчанов А.Ю. Системное программное обеспечение: Учебник для вузов / А.Ю. Молчанов. – СПб.: Питер, 2010. – 400 с.

9. Сидоров М.О. Якість програмного забезпечення та тестування: Підручник / М.О. Сидоров, М.А. Безверха. – К.: НАУ, 2010. – 282 с.

**Паршукова Л.**

*Уманський державний педагогічний університет  
імені Павла Тичини*

## ОСОБЛИВОСТІ РОЗРОБКИ ДИДАКТИЧНИХ МАТЕРІАЛІВ ІНФОРМАТИКИ

Сучасні інформаційні технології дозволяють розробникам дидактичних матеріалів оперувати значним комплексом вербальних і невербальних засобів. Ці засоби дозволяють створювати естетичні, захоплюючі, пізнавальні, проблемні матеріали і тим самим підвищити мотивацію і пізнавальний інтерес учнів. Така психолого-педагогічна складова дидактичного матеріалу створює спонукальні мотиви до поглибленого вивчення того чи іншого питання та спрямована на привернення уваги учня, підтримку пізнавального інтересу, активізацію його мислення.

Оскільки, у завдання вчителя входить допомога учневі найбільш повно оволодіти знаннями і використовувати їх у вирішенні практичних завдань, ми ставимо перед собою завдання розробити такий дидактичний матеріал, який в повній мірі забезпечив розв'язання такого завдання.

Дидактичні матеріали з інформатики призначені для:

- самостійного оволодіння учнями матеріалом, формування умінь працювати з джерелами інформації;
- активізації пізнавальної діяльності учнів;
- формування умінь самостійно осмислювати та засвоювати новий навчальний матеріал;
- контролю зі зворотним зв'язком, з діагностикою помилок за результатами діяльності та оцінки результатів;
- самоконтролю і самокорекції;
- тренування в процесі засвоєння навчального матеріалу;
- посилення мотивації навчання;
- розвитку наочно-образного, теоретичного та логічного мислення;