

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ**

ЯКІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕСТУВАННЯ

**Методичні рекомендації
до самостійної роботи студентів
спеціальності 121 "Інженерія програмного забезпечення"
першого (бакалаврського) рівня**

**Харків
ХНЕУ ім. С. Кузнеця
2019**

УДК 004.05(07.034)

Я45

Укладач Ю. І. Скорін

Затверджено на засіданні кафедри інформаційних систем.
Протокол № 1 від 27.08.2018 р.

Самостійне електронне текстове мережеве видання

Якість програмного забезпечення та тестування [Електрон-
Я45 ний ресурс] : методичні рекомендації до самостійної роботи студентів спеціальності 121 "Інженерія програмного забезпечення" першого (бакалаврського) рівня / уклад. Ю. І. Скорін. – Харків : ХНЕУ ім. С. Кузнеця, 2019. – 48 с.

Подано в систематизованому вигляді методичні рекомендації до самостійної роботи студентів, згідно з тематичним планом навчальної дисципліни. Уміщено загальні положення щодо організації самостійної роботи студентів, методику підготовки до усного опитування, виконання практичних завдань, поточного тестового контролю.

Рекомендовано для студентів спеціальності 121 "Інженерія програмного забезпечення" першого (бакалаврського) рівня.

УДК 004.05(07.034)

© Харківський національний економічний
університет імені Семена Кузнеця, 2019

Вступ

Стрімкий розвиток програмної індустрії у світі обумовлює дедалі більш жорсткі вимоги до якості створюваних програмних продуктів, програмних систем, компонентів повторного використання та ресурсні обмеження їхнього досягнення. Для задоволення цих вимог у процесі розроблення програмних продуктів передбачено спеціальний вид діяльності, названий гарантуванням якості.

Щораз більшу актуальність цієї проблематики обумовлено зростанням масштабу потенційних негативних наслідків помилок у програмних продуктах критичного призначення через поширення їхнього застосування в найбільш різноманітних галузях господарства; упровадженням індустріальних технологій повторного використання готових ресурсів, для яких необхідно гарантований рівень якості; розвитком нових організаційних форм розподіленого розроблення і використання програмних продуктів; зростанням витрат на опрацювання помилок у продуктах, залежно від часу між моментами їхнього внесення і виявлення та обсягом проекту; розвитком технологічних підходів до побудови моделей процесів тестування і забезпечення надійності з бажаними властивостями та необхідного математичного апарату.

Тому сучасні економічні умови господарювання потребують від фахівців, незалежно від їхньої спеціалізації, усебічного використання новітніх інформаційних технологій, комп'ютеризованих засобів збирання, оброблення та видавання необхідної інформації, створення на їхній основі сучасного програмного забезпечення, із метою підвищення якості та оперативності економічних розрахунків, зробити значно більш ефективним процес обґрунтування економічних рішень тощо.

У цьому контексті найбільш актуальним стає питання забезпечення якості розробленого програмного продукту, а саме тестування, яке і є одним із найбільш ефективних способів підвищення якості розроблення програмного продукту, що входить до системи забезпечення якості програмного продукту.

Якість програмних продуктів – це сукупність властивостей програмного забезпечення, що забезпечують її спроможність задовольняти встановлені, передбачувані або очікувані потреби, відповідно до призначення.

Своєю чергою, якість програмного продукту характеризується набором властивостей, що визначають, наскільки продукт задовольняє вимоги таких зацікавлених сторін, як: замовник продукту, спонсор, кінцевий користувач, розробники й тестери продукту, інженери підтримки, співробітники відділів маркетингу, навчання та продажів.

Кожен з учасників може мати різне уявлення про продукт і про те, наскільки він відповідає їхнім вимогам, тобто про те, наскільки висока якість продукту.

Таким чином, поставлення завдання забезпечення якості програмного продукту переходить у завдання визначення зацікавлених осіб, їхніх критеріїв якості, а потім знаходження оптимального рішення, що задовольняє цим критеріям.

Тестування є одним із найбільш усталених способів забезпечення якості розроблення програмного забезпечення та входить до набору ефективних засобів сучасної системи забезпечення якості програмного продукту.

Із технічного погляду тестування полягає у виконанні програми на деякій множині вихідних даних і звірванні здобутих результатів із заздалегідь відомими (еталонними), із метою встановлення відповідності різних властивостей і характеристик програмного продукту замовленим властивостям.

Як одна з основних фаз процесу розроблення програмного продукту, тестування характеризується досить великим внеском у сумарну трудомісткість розроблення продукту, унаслідок цього найбільшого ефекту у зниженні трудомісткості може бути досягнуто, насамперед, на фазі тестування.

Тому основні внески в автоматизацію або генерацію коду слід здійснювати, насамперед, на цій фазі. Хоча в сучасному індустріальному програмуванні автоматизація тестування є досить поширеною практикою, водночас технологія верифікації вимог та специфікацій поки робить лише свої перші кроки.

Завданням найближчої перспективи є рух у бік такого розподілу трудомісткості, щоб сумарна ціна виявлення більшості дефектів прагнула до мінімуму за рахунок виявлення переважного числа на найбільш ранніх фазах розроблення програмного продукту.

Навчальну дисципліну зараховано до групи базових дисциплін для студентів спеціальності 121 "Інженерія програмного забезпечення" першого (бакалаврського) рівня. Вона є невід'ємною частиною циклу комп'ютерних дисциплін, необхідних фахівцям-аналітикам із:

вивчення понять, принципів, методології та технологій тестування програмних продуктів (ПП);

засвоєння основних понять і визначень із галузі тестування, критеріїв вибору тестів;

огляду різновидів тестування;

аналізу особливостей процесу й технології тестування;

набуття навичок у застосуванні сучасних інформаційних технологій для аналізу та тестування інформаційних систем;

створення звітної тестової документації.

Метою викладання цієї навчальної дисципліни є засвоєння студентами основних понять і визначень із галузі тестування, критеріїв вибору тестів, огляд різновидів тестування, аналіз особливостей процесу й технології індустріального тестування, набуття навичок у застосуванні сучасних інформаційних технологій для аналізу та тестування інформаційних систем, створенні звітної тестової документації.

Основні поняття забезпечення якості та інструментарій тестування програмного забезпечення викладають студентам спеціальності 121 "Інженерія програмного забезпечення" в межах навчальної дисципліни "Якість програмного забезпечення та тестування". Вивчення цієї навчальної дисципліни ґрунтується на знаннях та вміннях, які студенти здобули під час вивчення таких навчальних дисциплін: "Основи алгоритмізації", "Методи і засоби комп'ютерних інформаційних технологій", "Сучасні засоби програмування", "Технології програмування та створення програмних продуктів".

Для досягнення мети поставлено такі основні завдання:

засвоєння основних термінів і визначень із галузі тестування, фаз та технологій тестування, головних проблем і завдань тестування;

вивчення структурних, функціональних, стохастичних, мутаційних критеріїв вибору тестів;

вивчення модульного, інтеграційного, системного, регресійного тестування, автоматизації тестування, витрат тестування;

планування тестування, підходів до розроблення тестів, аналіз особливості ручного розроблення й генерації тестів;

документування етапів та результатів тестування, аналізування методів відбору тестів, оцінювання їхньої ефективності.

Об'єктом навчальної дисципліни є процеси, що відображають різні аспекти забезпечення якості програмних продуктів із використанням сучасних методів та засобів тестування.

Предметом цієї навчальної дисципліни є основи теорії забезпечення якості програмних продуктів за допомогою сучасних методів і засобів тестування, сучасні інформаційні технології для тестування інформаційних систем, методики розроблення та створення тестової документації для оцінювання якості програмного продукту.

Вивчення навчальної дисципліни спрямовано на набуття студентами компетентностей щодо сучасних методів і засобів забезпечення якості програмного продукту шляхом використання сучасної технології індустріального тестування інформаційних систем та їхніх компонентів.

У результаті вивчення навчальної дисципліни студент повинен *знати*:

основні поняття забезпечення якості програмних продуктів;

особливості індустріального підходу в оцінюванні якості програмного продукту засобами тестування;

основні функціональні фази тестування й основні завдання та проблеми тестування;

критерії вибору тестів;

критерії оцінювання покриття проекту;

різновиди тестування;

методику планування тестування, підходи до розроблення тестів, особливості ручного розроблення й генерації тестів;

структуру тестового набору для автоматизованого тестування;

структуру документів "Тестовий план", "Тестова процедура", "Тестовий звіт";

методику опису тестових наборів та тестових звітів;

уміти:

аналізувати особливості процесу й технології тестування;

планувати тестування;

вибирати кінцевий набір тестів;

забезпечувати розроблення тестів з урахуванням особливостей ручного розроблення й генерації тестів і з використанням засобів автоматизації тестового циклу;

виконувати тестування на основі специфікацій та сценаріїв;

розробляти звітну тестову документацію.

У процесі викладання навчальної дисципліни основну увагу приділяють оволодінню студентами професійними компетентностями, наведеними в табл. 1.

Таблиця 1

**Професійні компетентності, яких набувають студенти
після вивчення навчальної дисципліни**

Назви компетентностей	Складові частини компетентностей
Здатність до організації роботи з використанням сучасних засобів забезпечення якості програмного забезпечення шляхом використання сучасної технології індустріального тестування інформаційних систем і їхніх компонентів та розроблення заходів щодо підвищення їхньої ефективності	Розробляти рекомендації та заходи, організувати та професійно брати участь у реалізації IT-рішень у діяльності щодо підвищення ефективності індустріального тестування
Уміння розробляти ефективні системи мотивації з використання інформаційних технологій проведення індустріального тестування інформаційних систем та їхніх компонентів	Розробляти систему мотивації персоналу щодо здійснення заходів зі створення комп'ютерних програм та їхнього впровадження в галузі індустріального тестування інформаційних систем
Здатність до пошуку сфер діяльності, у яких можуть бути використані сучасні інформаційні технології тестування інформаційних систем для більш ефективної діяльності бізнесу	Розробляти проекти, які забезпечують підвищення ефективності діяльності в бізнесі, завдяки підвищенню ефективності забезпечення якості програмного забезпечення шляхом використання сучасної технології індустріального тестування інформаційних систем та їхніх компонентів
Здатність розробляти технології створення та впровадження засобів комп'ютерної системи оброблення інформації в діяльності бізнесу	Розробляти та реалізовувати алгоритми забезпечення якості програмного забезпечення шляхом використання тестування інформаційних систем та їхніх компонентів

1. Самостійна робота студентів

Необхідним елементом успішного засвоєння навчального матеріалу цієї дисципліни є самостійна робота студентів (СРС), із метою поглибленого вивчення окремих теоретичних положень навчальної дисципліни.

Практично всі види аудиторних навчальних занять, і навіть лекції, потребують від студентів попередньої самостійної роботи. Крім того, у закладі вищої освіти (ЗВО) їм надають певну свободу вибору.

Таким чином, набагато більше значення має не те, скільки із зазначених джерел вивчив студент, а наскільки глибоко він засвоїв вивчений матеріал.

Крім того, через брак часу не всі теми навчальної дисципліни докладно розглядають на практичних заняттях. Деякі теми викладач не виносить навіть на лекцію, указуючи лише, до яких джерел студенти мають звернутися. Такий навчальний матеріал залишають для самостійного вивчення.

Тобто позааудиторна самостійна робота студента у ЗВО не менш важлива, ніж обов'язкові навчальні заняття. Її успішність багато в чому визначено тим, наскільки вміло, раціонально сам студент зможе організувати свої індивідуальні заняття, наскільки регулярними та своєчасними вони будуть.

СРС – це є форма організації навчального процесу, за якої заплановані завдання виконуються студентом самостійно під методичним керівництвом викладача.

Мета СРС – це є засвоєння в повному обсязі навчальної програми та формування у студентів загальних і професійних компетентностей, які відіграють суттєву роль у становленні майбутнього фахівця вищого рівня кваліфікації.

Самостійну роботу студентів реалізують у різних видах. Під час виконання будь-яких видів самостійної роботи надають можливість отримати консультацію викладача. Правильна організація СРС дає викладачеві можливість забезпечити вивчення найбільш складних питань програми з навчальної дисципліни.

Навчальний час, відведений для самостійної роботи студентів денної форми навчання, визначають навчальним планом навчальної дисципліни.

Під час самостійної роботи студент має бути активним учасником навчального процесу, навчитися свідомо ставитися до оволодіння теоретичними та практичними знаннями, вільно орієнтуватися в інформаційному

просторі, брати на себе індивідуальну відповідальність за якість власної професійної підготовки.

СРС охоплює:

опрацювання лекційного матеріалу;

доопрацювання лекційного матеріалу;

опрацювання та вивчення рекомендованої літератури, основних термінів і понять за темами дисципліни;

підготовку до лабораторних занять за допомогою методичних рекомендацій та рекомендованої літератури;

поглиблене опрацювання окремих лекційних тем або питань за допомогою лекційного матеріалу та рекомендованої літератури;

пошук (підбір) та огляд літературних джерел на задану проблематику дисципліни;

аналітичний розгляд наукової публікації;

контрольну перевірку студентами власних знань за запитаннями для самодіагностики;

підготовку до контрольних робіт (тестування) та інших форм поточного контролю за допомогою лекційного матеріалу й рекомендованої літератури;

комплексну систематизацію вивченого матеріалу, із метою підготовки до заліку.

Необхідним елементом успішного засвоєння матеріалу навчальної дисципліни є самостійна робота студентів із вітчизняною та зарубіжною спеціальною економічною літературою, нормативними актами з питань державного регулювання економіки, статистичними матеріалами.

Основні види самостійної роботи, запропоновані студентам для засвоєння теоретичних знань із навчальної дисципліни "Якість програмного забезпечення та тестування", наведено в табл. 2.

Таблиця 2

Завдання для самостійної роботи студентів

Назви тем	Зміст самостійної роботи	Література
1	2	3
Змістовий модуль 1. Основи тестування програмного забезпечення		
<i>Тема 1.</i> Тестування як спосіб забезпечення якості програмного продукту	Вивчення лекційного матеріалу; огляд теоретичного матеріалу та підготовка доповіді "Поняття забезпечення якості програмного забезпечення"	Основна: [2; 5; 7; 18]. Додаткова: [27 – 29]

1	2	3
<i>Тема 2.</i> Тестування на етапі розроблення програмного продукту	Вивчення лекційного матеріалу; підготовка до лабораторного заняття, огляд теоретичного матеріалу	Основна: [2; 13; 25]. Додаткова: [27 – 34]
<i>Тема 3.</i> Тестування на етапах планування та проектування програмного продукту	Вивчення лекційного матеріалу; огляд теоретичного матеріалу та підготовка доповіді "Критерії вибору тестів"	Основна: [2; 3; 9; 26]
<i>Тема 4.</i> Тестування на стадії кодування	Вивчення лекційного матеріалу; підготовка до лабораторного заняття	Основна: [2; 9; 12]
<i>Тема 5.</i> Програмні помилки	Вивчення лекційного матеріалу; підготовка до лабораторного заняття і контрольної роботи та підготовка доповіді "Програмні помилки"	Основна: [9; 12; 26]. Додаткова: [27 – 29]
Змістовий модуль 2. Організація процесу тестування програмного забезпечення		
<i>Тема 6.</i> Система відстеження дефектів. Планування тестування	Вивчення лекційного матеріалу; огляд теоретичного матеріалу та підготовка доповіді "Особливості інтеграційного тестування"	Основна: [2; 9; 12]. Додаткова: [27 – 29]
<i>Тема 7.</i> Документування тестових матеріалів	Вивчення лекційного матеріалу; підготовка до лабораторного заняття та до контрольної роботи	Основна: [9; 12; 26]. Додаткова: [27 – 29]
<i>Тема 8.</i> Автоматизація тестування	Вивчення лекційного матеріалу; підготовка до лабораторного заняття, огляд теоретичного матеріалу	Основна: [2; 9; 25]. Додаткова: [27 – 30]
<i>Тема 9.</i> Особливості індустріального тестування	Вивчення лекційного матеріалу; огляд теоретичного матеріалу та підготовка доповіді "Особливості індустріального тестування"	Основна: [9; 12; 26]. Додаткова: [27 – 29]
<i>Тема 10.</i> Проблеми якості програмного забезпечення та практичні рекомендації	Вивчення лекційного матеріалу; огляд теоретичного матеріалу та підготовка доповіді "Документування індустріального тестування"	Основна: [5; 7; 23]. Додаткова: [27 – 29]

Підготовка професіоналів для роботи в ІТ компаніях потребує принципово інших методів, порівняно із традиційними, необхідність у їхньому розробленні стає все більш гострою.

Для вирішення зазначеної проблеми доцільно використовувати в ході організації самостійної роботи студентів підходи, застосовані у промисловому розробленні програмного забезпечення та асоційовані на сьогодні з поняттям "тестування програмних продуктів".

У цьому підході застосовують активні форми навчання, а саме імітують процес розроблення програмного забезпечення, прийнятий в ІТ галузі, тим самим процес навчання наближають до умов реального ринку праці.

Для опису цієї моделі процесу навчання необхідно ввести такі поняття:

1. **Учасник процесу навчання.** Учасниками процесу навчання є студент і викладач.

2. **Роль.** Під роллю розуміють сукупність дій і функцій, що виконуються учасниками процесу. Студент і викладач у процесі навчання з використанням методів програмної інженерії можуть перебувати в різних ролях, причому одна й та сама роль може виконуватися як студентом, так і викладачем.

Ураховуючи, що основним об'єктом у цій моделі є завдання щодо створення програмного продукту, над якою відбувається колективна робота, необхідно ввести такі ролі учасників цього колективного освітнього процесу.

1. **Автор.** До завдань автора входить формулювання завдання для розроблення студентом.

2. **Модератор.** До завдань модератора входить призначення завдання для виконавців, контроль за процесом роботи над завданням і прийняття остаточного рішення про її результат.

3. **Кодувальник.** Завдання кодувальника – написання програми для вирішення призначеного йому завдання.

4. **Інспектор.** Інспектор робить аналіз коду розробленої програми щодо її коректності та якості кодування, адекватності застосовуваних програмних рішень. Інспектор оголошує один із трьох можливих вердиктів:

а) прийняти завдання без зауважень. Це означає, що інспектору абсолютно зрозумілі всі програмні рішення, код написано правильно, текст програми забезпечено всіма необхідними коментарями;

б) прийняти завдання за умови усунення зауважень. Це означає, що програму загалом написано правильно, однак деякі моменти залишаються незрозумілими або некоректними. Інспектор формулює свої

зауваження і перевіряє завдання знову після їхнього усунення кодувальником. Процес усунення зауважень триває до тих пір, поки інспектор не вирішить "Прийняти без зауважень";

в) відхилити завдання. Це означає, що програма не виконує необхідних функцій або кількість зауважень настільки велика, що її не може бути прийнято без суттєвого доопрацювання.

5. *Тестер*. До завдання тестера входить розроблення тесту, який дозволить перевірити якість написання програми шляхом пошуку в ній помилок, і власне перевірка. Роботу тестер починає після того, як інспектор оголосив вердикт "Прийняти без зауважень". У цьому разі тестер не повинен аналізувати код програми, він лише має запускати її на сценаріях, адекватних умовам завдання.

3. Робочий продукт. Результат роботи учасника процесу в тій чи тій ролі.

Робочими продуктами розглянутої моделі процесу навчання є:

в автора – формулювання завдання;

у модератора – оцінювання роботи виконавців;

у кодувальника – код програми (текст);

в інспектора – зауваження і набір рекомендацій із доопрацювання коду;

у тестера – сценарій тестів.

Послідовність виконання ролей у процесі роботи над програмою:

автор формулює завдання, для вирішення якого необхідно написати програму, і передає її кодувальнику;

модератор призначає ролі;

кодувальник пише програму та передає її інспектору;

інспектор аналізує програмний код і приймає своє рішення. Якщо код він приймає без зауважень, то програмний продукт передає тестеру. Якщо в інспектора є зауваження, він повідомляє їх кодувальнику. Далі відбувається взаємодія кодувальника з інспектором щодо усунення зауважень до прийняття рішення "Прийняти без зауважень";

тестер розробляє сценарій і перевіряє якість програмного продукту;

модератор приймає остаточне рішення про якість програмного продукту й оцінює результати роботи.

Можливо кілька варіантів розподілу ролей між учасниками процесу. У табл. 3 наведено розподіл у межах традиційних форм процесу навчання.

**Розподіл ролей між учасниками освітнього процесу
за умови традиційної форми навчання**

Ролі	Учасники	Функції
Автор	Викладач	Формулювання завдання
Модератор	Викладач	Розподіл завдань між студентами (видавання завдань)
Кодувальник	Студент	Написання програмного коду
Інспектор	Викладач	Перевірка коду програми, оцінювання роботи студента
Тестер	Викладач	Запуск програми та її тестування, оцінювання роботи студента

У традиційному варіанті ролі автора, модератора, інспектора та тестера виконує викладач, а студент – лише роль кодувальника.

Застосування такого підходу на сьогодні виправдане лише для навчання студентів непрофільних, не пов'язаних безпосередньо з інформаційними технологіями, напрямів підготовки студентів.

У табл. 4 наведено розподіл ролей для першого варіанта пропонуваної технології навчання.

Таблиця 4

**Розподіл ролей між учасниками освітнього процесу,
пропонується технологія навчання з використанням методів
програмної інженерії (варіант 1)**

Ролі	Учасники	Функції
Автор	Викладач	Формулювання завдання
Модератор	Викладач	Розподіл завдань між студентами, визначення ролі кожного зі студентів, оцінювання роботи студента
Кодувальник	Студент 1	Написання програмного коду
Інспектор	Студент 2	Перевірка коду програми, формулювання зауважень із поліпшення і доопрацювання
Тестер	Студент 3	Розроблення сценаріїв тестування, запуск програми та її тестування

У варіанті, наведеному в табл. 4, викладач виконує ролі автора й модератора, а ролі кодувальника, інспектора та тестера виконують студенти.

Викладач формулює завдання на ту чи ту тему, розподіляє ролі між студентами, контролює процес їхньої роботи й оцінює результат.

Студент-кодувальник пише програмний код і передає його студентові-інспектору.

Інспектор аналізує поданий код, формулює свої зауваження та (у разі потреби) повертає його кодувальнику на доопрацювання. За умови відсутності зауважень програму передає студентові-тестеру.

Тестер розробляє сценарій тесту і "проганяє" через нього програму з формулюванням висновку типу "пройдено або не пройдено" за кожним пунктом сценарію.

Після закінчення роботи тестера викладач дає оцінку кожному учасникові.

У тому разі, якщо для групи студентів реалізують поглиблене вивчення програмування, можливий другий варіант розподілу ролей учасників (табл. 5).

Таблиця 5

**Розподіл ролей між учасниками освітнього процесу,
пропонована технологія навчання з використанням методів
програмної інженерії (варіант 2)**

Ролі	Учасники	Функції
Автор	Студент 1	Формулювання завдання
Модератор	Викладач	Визначення ролі кожного зі студентів, розподіл завдань між студентами, оцінювання роботи студента
Кодувальник	Студент 2	Написання програмного коду
Інспектор	Студент 3	Перевірка коду програми, формулювання зауважень з поліпшення і доопрацювання
Тестер	Студент 4	Розроблення сценаріїв тестування, запуск програми та її тестування

У цьому варіанті роль автора виконують студенти.

Застосування описаної технології у процесі вивчення цієї навчальної дисципліни, а саме під час організації самостійної роботи студентів, дозволяє підвищити ефективність навчального процесу з погляду набуття студентами компетентностей у сфері тестування програмних продуктів.

Крім цього, її використання дозволяє підвищити мотивацію студентів до вивчення навчальної дисципліни, познайомити їх із прийомами колективної роботи, які застосовують у професійному середовищі, та забезпечити прозорість виставлення оцінок із навчальної дисципліни [23].

2. Методичні рекомендації щодо підготовки до лекційних занять, роботи з рекомендованою літературою

Головна мета в період підготовки до лекційних занять – навчитися методам самостійної розумової праці, свідомо розвивати свої творчі здібності та оволодівати навичками у творчій роботі.

Запис лекцій – це складний вид вузівської аудиторної роботи. Уважне слухання і конспектування лекцій передбачає інтенсивну розумову діяльність студента. Короткі записи лекцій, їхнє конспектування допомагає засвоїти навчальний матеріал. Конспект є корисним тоді, якщо записано основну інформацію і виконано це самим студентом. Не треба прагнути записати дослівно всю лекцію. Таке "конспектування" завдає більше шкоди, ніж користі. Запис лекцій рекомендують вести по можливості власними формулюваннями. Бажано здійснювати запис на одній сторінці, а наступну залишати для опрацювання навчального матеріалу самостійно. Конспект лекції краще розподіляти на пункти, параграфи тощо. Цьому більшою мірою будуть сприяти пункти плану лекції, запропоновані викладачем.

Але не менш важливим є вивчення конспекту лекцій, його доповнення. Працюючи з конспектом лекцій, завжди необхідно використовувати не тільки підручник, але й ту літературу, яку додатково рекомендував лектор. Саме така серйозна, копітка робота з лекційним матеріалом дозволить глибоко оволодіти знаннями.

Алгоритм виконання завдання зі збагачення і доповнення лекційного матеріалу в ході самостійної роботи:

усвідомте та конкретизуйте для себе мету завдання, наприклад, доопрацювати тему лекції, підготуватися до контрольної роботи, з'ясувати підстави неоднозначних підходів різних авторів до одного й того ж явища, поняття тощо;

відтворіть основні положення прочитаної лекції викладачем із досліджуваного питання;

установіть зв'язок між змістом лекції та проблемою, яка була розглянута у процесі вивчення нового літературного джерела;

прочитайте літературне джерело, усвідомте його зміст, звернувши увагу на закладені в ньому світоглядні ідеї, сутність і значення;

з'ясуйте методологічну сутність, педагогічний зміст роботи, мету та завдання, запропоновані методи їхнього досягнення, можливий результат;

складіть план матеріалу, із яким працюєте, і, відповідно до нього, законспекуйте рекомендований літературним джерелом матеріал;

конспектуючи літературне джерело, зобразіть у вигляді схем, діаграм і таблиць деякі його положення, факти, статистичні дані, що допоможе краще засвоїти досліджуваний матеріал;

підготуйте відповіді на поставлені викладачем запитання, усно або письмово відтворіть основні положення та факти;

запишіть питання, які хотілося б запропонувати для обговорення у групі або які викликають сумніви.

Уміння самостійно працювати з літературою – ще одне з важливих умінь, яке повинен набути студент під час навчання у ЗВО. Дійсно, на сьогодні накопичено людством таку величезну кількість інформації, яка до того ж постійно збільшується, що людина не в змозі засвоїти в повному обсязі навіть у межах однієї галузі знання. Тому набагато важливіше навчитися орієнтуватися в цьому морі найрізноманітніших відомостей, швидко знаходити джерела необхідної інформації, тобто насамперед знати, що, де і як шукати.

Самостійна робота з підручниками, навчальними посібниками, науковою, довідковою і популярною літературою, матеріалами періодичних видань та Інтернету, статистичними даними є ефективним методом здобуття знань, дозволяє значно активізувати процес оволодіння інформацією, сприяє глибокому засвоєнню матеріалу, що вивчають, формує у студентів своє ставлення до конкретної проблеми. Глибшому розкриттю питань сприяє знайомство з додатковою літературою, рекомендованою викладачем до кожної теми практичного заняття, що дозволяє студентам виявити свою індивідуальність під час виступу на цих заняттях, широкий спектр думок із досліджуваної проблеми.

3. Методичні рекомендації щодо підготовки до виконання практичних завдань

Практичні види роботи становлять ряд завдань із навчальної дисципліни для самостійного виконання під час підготовки до занять та безпосередньо в ході занять. У середньому виконання практичного завдання, залежно від складності обраної теми та особливостей організації студентом своєї самостійної роботи, становить від 30 до 90 хвилин.

Практичні завдання виконують на заняттях. Оцінку за виконання практичних завдань ураховують у роботі на заняттях. Додаткові бали (від одного до двох) за самостійну підготовку до практичних занять студент може набрати за умови якісного виконання самих завдань, а також за підготовку есе за матеріалами, які не ввійшли до лекційного курсу.

У процесі підготовки до виконання практичних завдань студенту необхідно опрацювати теоретичний матеріал із досліджуваної теми, наведений у лекційному курсі, методичні рекомендації до виконання практичних робіт, виконати приклади практичних завдань, що містяться в робочій навчальній програмі, підготувати есе на задану тему.

Есе на задану тему – це викладання думок на тему у вільній формі, тобто есе – це літературна форма, невеликий прозовий текст, що виражає індивідуальну думку автора. Жанр есе передбачає свободу творчості. Головне в есе – особистість автора, його думки, почуття, ставлення до світу. Це основне завдання твору. Однак слід пам'ятати, що, незважаючи на свободу творчості, писати в жанрі есе нелегко, оскільки потрібно знайти оригінальну ідею (навіть на традиційному матеріалі), виразити нестандартний погляд на будь-яку проблему.

Стиль есе відрізняється образністю та парадоксальністю. Для передавання особистісного сприйняття, освоєння світу автор залучає численні приклади, проводить паралелі, підбирає аналогії, використовує всілякі асоціації.

Специфіка жанру есе полягає в тому, що якщо ви хочете сказати щось своє, нове, нестандартне, то жанр есе – це саме ваш жанр.

Основною відмінною рисою есе є його стислість, і хоча обсяг може коливатися від однієї до п'ятдесяти сторінок, переважно воно займає від десяти до двадцяти сторінок. Есе має певну внутрішню єдність, яку,

зазвичай, важко зберегти у великих прозових текстах, що виражають авторське бачення.

Написані есе студенти здають викладачеві на перевірку, основною формою якої є рецензування. Можна застосовувати й інші форми перевірки, наприклад, взаєморецензування робіт студентами або захист есе у формі усної доповіді на практичних видах занять [65].

4. Перелік тем есе

Змістовий модуль 1

Основи тестування програмного забезпечення

Тема 1. Тестування як спосіб забезпечення якості програмного продукту

1. Складові частини якості програмного забезпечення: надійність, супровід, практичність, ефективність, мобільність, функціональність.
2. Стандарт якості ISO – світовий стандарти якості ПП.
3. Історична довідка про розвиток інформаційних технологій забезпечення якості програмного забезпечення шляхом тестування.
4. Мінімальний набір вимог до управління якістю.
5. Вимоги до контролю за продукцією.
6. Характеристики якості програмного продукту.
7. Методи визначення показників якості.
8. Метрики програмного продукту та їхні складові частини.

Тема 2. Тестування на етапі розроблення програмного продукту

1. Склад групи тестування.
2. Права й обов'язки членів групи тестування.
3. Критерії вибору тестів.

Тема 3. Тестування на етапах планування та проектування програмного продукту

1. Класифікація тестів за рівнем автоматизації.
2. Порівняльний аналіз ручного й автоматизованого тестування.
3. Інструментарій для автоматизованого тестування.

Тема 4. Тестування на стадії кодування

1. Стратегії тестування.
2. Тестування за стратегією "скляної скриньки".
3. Тестування за стратегією "чорної скриньки".

Тема 5. Програмні помилки

1. Методи та засоби оброблення програмних помилок.
2. Перенавантаження.
3. Вплив причини "занадто велика кількість усіх можливих комбінацій вхідних даних".
4. Вплив причини "занадто велика кількість усіх можливих послідовностей виконання коду програми".

Змістовий модуль 2

Організація процесу тестування програмного забезпечення

Тема 6. Система відстеження дефектів. Планування тестування

1. Питання прогнозування помилок.
2. Критерії тестування вимог.
3. Система відстеження дефектів.
4. Планування тестування.

Тема 7. Документування тестових матеріалів

1. Призначення та структура плану тестування.
2. Компоненти плану тестування.
3. Підсумковий звіт тестування.
4. Методи планування тестування документації.
5. Документування тестових матеріалів.

Тема 8. Автоматизація тестування

1. Автоматизація тестування.
2. Видатки тестування.
3. Комбінація методів тестування.
4. Виконання тестів.
5. Ручне тестування.

6. Автоматизоване тестування.
7. Порівняння ручного й автоматизованого тестування.

Тема 9. Особливості індустріального тестування

1. Сутність визначення "індустріальний підхід".
2. Особливості індустріального тестування програмного продукту.
3. Тестування специфікації.
4. Тестування сценаріїв.
5. Тестові метрики.
6. Видатки на тестування програмних продуктів.

Тема 10. Проблеми якості програмного забезпечення та практичні рекомендації

1. Документування індустріального тестування програмного продукту.
2. Оцінювання індустріального тестування.
3. Характеристики процесу документування тестових матеріалів.
4. Види звітів про виконання тестування.
5. Документування та життєвий цикл дефекту програмного продукту.
6. Тестовий звіт.

5. Методичні рекомендації щодо підготовки до поточного тестового контролю (контрольної роботи)

Своєчасно і якісно виконані письмові роботи є дієвою й конструктивною формою зв'язку індивідуальності студента з викладачем.

Зміст роботи дозволяє виявити ступінь розуміння та засвоєння студентом навчального матеріалу, його вмінь самостійно працювати в різних аспектах освітнього процесу рівня ЗВО.

Успіх і якість виконання письмових робіт обумовлено обізнаністю студента в навчальному плані (за курсами навчання), у змісті освітніх програм із навчальних дисциплін, його вміння доцільно планувати й організовувати самостійну роботу, а так само від наявності ясних, чітких рекомендацій і вимог педагогів, що висувають до змісту виконуваної студентом роботи.

Письмову роботу розглядають як основну навчальну діяльність студента, що дає можливість творчо, індивідуально й оригінально розкрити та реалізувати потенціал майбутнього.

Успішне виконання тестового контролю є необхідною умовою підсумкової позитивної оцінки, відповідно до рейтингової системи навчання.

Тестові завдання підготовлено на основі лекційного матеріалу, підручників та навчальних посібників із цієї навчальної дисципліни.

Виконання таких завдань надає студентам можливість самостійно контролювати рівень своїх знань, виявляти "прогалини" у знаннях і вживати заходів щодо їхньої ліквідації. Запропоновані тестові завдання охоплюють вузлові питання теоретичних і практичних основ із цієї навчальної дисципліни.

Форма викладу тестових завдань дозволяє закріпити та відновити в пам'яті засвоєний матеріал.

Для формулювання завдань використано закрити форму. У студента є можливість вибору правильної відповіді або кількох правильних відповідей із запропонованих варіантів.

Для виконання тестових завдань студенти повинні вивчити лекційний матеріал із теми, відповідні розділи підручників, навчальних посібників та інших літературних джерел.

Контрольні тестові завдання виконуються студентами на лабораторних заняттях.

Пробні тестові завдання містяться в робочій навчальній програмі навчальної дисципліни. Із ними доцільно ознайомитися під час підготовки до контрольного тестування.

Порядок здійснення поточного оцінювання знань студентів полягає в такому.

Поточний тестовий контроль (контрольну роботу) проводять два рази за семестр. Однією з основних і безперечних його переваг є мінімум часових витрат на визначення надійних підсумків контролю.

У ході тестування використовують електронні варіанти, які дозволяють визначити результати практично відразу після завершення тесту.

Тестування виконує три основні взаємопов'язані функції:

діагностичну;

навчальну;

виховну.

Діагностична функція полягає у виявленні рівня знань, умінь та навичок студента. Це основна функція тестування.

Навчальна функція тестування полягає в мотивуванні студента до активізації роботи із засвоєння навчального матеріалу.

Із метою посилення навчальної функції тестування, може бути використано такі додаткові заходи стимулювання студентів, як:

роздавання викладачем приблизного переліку питань для самостійної підготовки;

наявність у самому тесті підказок;

спільне опрацювання результатів тесту.

Виховну функцію виявляють у періодичності тестового контролю.

Це дисциплінує, організовує та спрямовує діяльність студентів, допомагає виявити й усунути "прогалини" у знаннях, формує прагнення розвинути свої здібності.

Тест передбачає запитання одиничного та множинного вибору щодо перевірки знань основних категорій навчальної дисципліни.

Максимальна кількість балів за контрольну роботу становить 12.

Розподіл балів має такий вигляд:

якщо студент дав правильну відповідь на 1 – 2 % запитань, то він набирає 1 бал;

якщо 3 – 10 % – 2 бали;

якщо 11 – 19 % – 3 бали;

якщо 20 – 28 % – 4 бали;

якщо 29 – 37 % – 5 балів;

якщо 38 – 46 % – 6 балів;

якщо 47 – 55 % – 7 балів;

якщо 56 – 64 % – 8 балів;

якщо 65 – 73 % – 9 балів;

якщо 74 – 82 % – 10 балів;

якщо 83 – 91 % – 11 балів;

якщо 92 – 100 % – 12 балів.

Приклад запитань одиничного вибору наведено на рис. 1, приклад множинного вибору наведено на рис. 2.

Вопрос:

Качество программного продукта - это:

Баллы за задание: Ограничение по времени: 0:02:00 Рисунок: + [иконки]

Укажите несколько вариантов ответа:

- 1 **точное соответствие разработанного программного продукта спецификации, при условии, что спецификация существует и она верна**
- 2 **соответствие разработанного программного продукта мировым стандартам**
- 3 **соответствие разработанного программного продукта нормам системы стандартов менеджмента качества ISO**
- 4 **точное соответствие разработанного программного продукта спецификации клиента**
- 5 **соответствие разработанного программного продукта стандартам ISO**

Рис. 1. Пример задания с вариантами ответа

Вопрос:

Выделяют следующие фазы процесса тестирования:

Баллы за задание: Ограничение по времени: 0:02:00 Рисунок:

Укажите несколько вариантов ответа:

- 1 **определение целей (требований к тестированию)**
- 2 **планирование**
- 3 **разработка тестов**
- 4 **выполнение тестов**
- 5 **оценка себестоимости тестирования**

Рис. 2. Пример задания множественного выбора

6. Перелік запитань тесту

1. Укажіть, хто із членів групи тестування відповідає за організацію процесу тестування:

- а) керівник групи тестування;
- б) тест-аналітик;
- в) тест-розробник;
- г) виконавець тестів.

2. Укажіть хто із членів групи тестування відповідає за розроблення тестових специфікацій та аналіз результатів тестування:

- а) керівник групи тестування;
- б) тест-аналітик;
- в) тест-розробник;
- г) виконавець тестів.

3. Виберіть правильне твердження:

- а) методи тестування описують у плані тестування;
- б) методи тестування описують у тестовій специфікації;
- в) методи тестування описують у тестових процедурах.

4. Укажіть, які документи використовують для розроблення плану тестування:

- а) технічне завдання;
- б) тестові специфікації;
- в) тестові процедури;
- г) підсумковий звіт тестування.

5. Які з критеріїв тестування використовують інформацію про структуру програми:

- а) структурні критерії;
- б) функціональні критерії;
- в) критерії стохастичного тестування;
- г) мутаційні критерії?

6. На якому етапі тестування застосовують структурні критерії:

- а) регресійному тестуванні;
- б) модульному тестуванні;

- в) інтеграційному тестуванні;
- г) системному тестуванні?

7. Які з критеріїв тестування забезпечують контроль за ступенем виконання вимог замовника у програмному продукті:

- а) структурні критерії;
- б) функціональні критерії;
- в) критерії стохастичного тестування;
- г) мутаційні критерії?

8. Яку модель тестування використовують у процесі функціонального тестування:

- а) чорний ящик;
- б) білий ящик?

9. Яку модель тестування використовують у ході модульного тестування:

- а) чорний ящик;
- б) білий ящик?

10. Яку модель тестування використовують у процесі інтеграційного тестування:

- а) чорний ящик;
- б) білий ящик?

11. Виявлення помилок, пов'язаних із роботою системи, загалом виконують таким способом:

- а) системним тестуванням;
- б) модульним тестуванням;
- в) інтеграційним тестуванням;
- г) тестуванням інтерфейсів;
- д) стрес-тестуванням;
- е) навантажувальним тестуванням.

12. До якого класу тестування можна зарахувати тестування інтерфейсів (екранів, кнопок тощо):

- а) системного тестування;

- б) функціонального тестування;
- в) GUI-тестування;
- г) UNIT-тестування;
- д) стрес-тестування;
- е) навантажувального тестування?

13. До якого класу тестів належить перевірка працездатності роботи системи:

- а) системного тестування;
- б) функціонального тестування;
- в) GUI-тестування;
- г) UNIT-тестування;
- д) стрес-тестування;
- е) навантажувального тестування?

14. Для перевірки працездатності окремих елементів системи використовують:

- а) системне тестування;
- б) функціональне тестування;
- в) GUI-тестування;
- г) UNIT-тестування;
- д) стрес-тестування;
- е) навантажувальне тестування.

15. Яке тестування необхідно виконати для перевірки працездатності програми в екстремальних умовах:

- а) системне тестування;
- б) функціональне тестування;
- в) GUI-тестування;
- г) UNIT-тестування;
- д) стрес-тестування;
- е) навантажувальне тестування?

16. Тестування характеристик програмного продукту, пов'язаних зі швидкістю оброблення даних, – це:

- а) системне тестування;
- б) функціональне тестування;

- в) GUI-тестування;
- г) UNIT-тестування;
- д) стрес-тестування;
- е) навантажувальне тестування?

17. Яке тестування слід виконувати в разі випуску нового релізу продукту:

- а) системне тестування;
- б) функціональне тестування;
- в) GUI-тестування;
- г) UNIT-тестування;
- д) стрес-тестування;
- е) навантажувальне тестування.

18. Якщо у вікні введення пароля розташовано кнопку підтвердження введення пароля "Реєстрація", то:

- а) це є помилкою;
- б) це не є помилкою.

19. Якщо у модальному вікні розташовано кнопку "Застосувати", то:

- а) це є помилкою;
- б) це не є помилкою.

20. Чи є помилкою наявність піктограм на застосованих кнопках:

- а) так;
- б) ні?

21. Якщо у вікні реєстрації користувача під час вибору в полі "Login" імені користувача зникає поле "Password", то:

- а) це є помилкою;
- б) це не є помилкою.

22. Якщо у процесі введення даних у полі введення відсутня підказка:

- а) це є помилкою;
- б) це не є помилкою.

23. Чи є обмеження на ширину поля зі списком:

- а) так;
- б) ні?

24. Якщо список містить понад сім елементів, водночас список не відсортовано, то:

- а) це є помилкою;
- б) це не є помилкою.

25. Якщо форма містить кілька радіокнопок, водночас жодну із кнопок не вибрано, то:

- а) це є помилкою;
- б) це не є помилкою.

26. Список містить понад 50 елементів. Якщо у цьому разі список відсортовано за частотою використання елементів, то:

- а) це є помилкою;
- б) це не є помилкою.

27. Тестують реєстраційна форма. Якщо після введення всіх даних натискають кнопку підтвердження введення і бачать повідомлення про неправильне введення даних у полі "Дата народження", то:

- а) це є помилкою роботи додатка;
- б) додаток працює правильно.

28. Тестують реєстраційну форму. Якщо після введення чисел у полі "Прізвище" відображається модальне вікно з повідомленням про помилку, то:

- а) це є помилкою роботи додатка;
- б) додаток працює правильно.

29. Тестують реєстраційну форму. Після введення останнього значення в поле введення форми натискають клавішу Enter. Якщо додаток почав оброблення форми, то:

- а) це є помилкою роботи додатка;
- б) додаток працює правильно.

30. Тестують головне меню додатка. Якщо в разі натиснення пункту меню "Довідники" відкривається форма довідників без заголовка, то:

- а) це є помилкою роботи додатка;
- б) додаток працює правильно.

31. Укажіть, хто із членів групи тестування відповідає за організацію роботи із замовником:

- а) керівник групи тестування;
- б) тест-аналітик;
- в) тест-розробник;
- г) адміністратор;
- д) виконавець тестів.

32. Укажіть, хто із членів групи тестування відповідає за розроблення тестової документації:

- а) керівник групи тестування;
- б) тест-аналітик;
- в) тест-розробник;
- г) виконавець тестів.

33. Укажіть, хто із членів групи тестування відповідає за розроблення автоматизованих тестів, установалення і супроводження тестового оточення:

- а) керівник групи тестування;
- б) тест-аналітик;
- в) тест-розробник;
- г) виконавець тестів.

34. Укажіть, хто із членів групи тестування відповідає за виконання тестів і реєстрацію помилок:

- а) керівник групи тестування;
- б) тест-аналітик;
- в) тест-розробник;
- г) адміністратор;
- д) виконавець тестів.

35. Укажіть, хто із членів групи тестування відповідає за роботу з документацією:

- а) керівник групи тестування;
- б) тест-аналітик.
- г) тест-розробник;
- д) виконавець тестів.

36. До функціональних вимог розроблення програмного забезпечення належать:

- а) вимоги розробника;
- б) бізнес-вимоги;
- в) вимоги користувача;
- г) опис цілей і завдань створюваної системи.

37. Вимоги до програмного забезпечення можуть розробляти щодо:

- а) продукту, його властивостей;
- б) процесу, окремих його стадій;
- в) процесу та його результату.

38. Функціональні вимоги до програмного забезпечення визначають:

- а) що необхідно зробити;
- б) як необхідно зробити;
- в) за допомогою кого необхідно зробити.

39. Нефункціональні вимоги до програмного забезпечення визначають:

- а) що необхідно зробити;
- б) як необхідно зробити;
- в) за допомогою кого необхідно зробити.

40. Функціональні вимоги до програмного забезпечення містять:

- а) бізнес-вимоги;
- б) бізнес-правила;
- в) бізнес-вимоги та власні вимоги.

41. До типів робочих продуктів (артефактів) у технології RUP належать:

- а) документ, модель, варіанти використання;
- б) варіант використання;
- в) план, документ, модель.

42. Дії в технології RUP розподіляють на:

- а) процедури;
- б) кроки;
- в) завдання.

43. Як шаблони в технології RUP можуть використовувати:

- а) розроблені робочі продукти;
- б) наявні прототипи;
- в) наявні продукти та прототипи проектування.

44. Ітерації розроблення програмного забезпечення, згідно з технологією RUP, вимірюють:

- а) складністю системи;
- б) тривалістю;
- в) кількістю ітерацій, тривалістю;
- г) масштабом організації, тривалістю.

45. CASE-засоби розроблення програмного забезпечення – це:

- а) інструментальні програмні засоби для проектування, формування вимог, кодування, тестування, документування та інших дій зі створення програмного забезпечення;
- б) інструментальні програмні засоби для аналізу вимог і проектування, текстові редактори, системи аналізу вихідних текстів;
- в) середовища візуального програмування та генерації вихідних кодів, компілятори, інтерпретатори;
- г) інструментальні програмні засоби для проектування, формування вимог, кодування.

46. До основних завдань бізнес-моделювання належать:

- а) опис бізнес-процесів, бізнес-функцій;
- б) опис бізнес-процесів і осіб, які беруть у них участь;
- в) опис бізнес-процесів, бізнес-функцій, бізнес-правил і осіб, які беруть у них участь;
- г) опис осіб, що беруть участь, та автоматизованої організаційної структури підприємства.

47. До основних цілей розроблення варіантів використання належать (вибрати аргументовано):

- а) визначення меж проєктованої системи, вимог, що до неї висувають;
- б) визначення основних осіб, які беруть участь, для реалізації бізнес-функцій;
- в) розроблення концептуальної моделі, визначення меж проєктованої системи, вимог та документації.

48. Варіант використання – це:

- а) тип сервісу, що надається системою конкретній особі, яка бере участь;
- б) безліч сервісів, що надаються системою особам, що беруть участь;
- в) варіант взаємодії із системою осіб, які беруть участь.

49. Структура життєвого циклу програмного продукту на основі методології RUP охоплює такі робочі процеси:

- а) управлінські, основні;
- б) основні, допоміжні;
- в) управлінські, допоміжні.

50. Процес ISO 12207 "Документування" належить до процесу:

- а) основного;
- б) допоміжного;
- в) організаційного.

51. Процес ISO 12207 "Забезпечення якості" належить до процесу:

- а) основного;
- б) допоміжного;
- в) організаційного.

52. Процес ISO 12207 "Управління проєктами" належить до процесу:

- а) основного;
- б) управління конфігурацією;
- в) організаційного.

53. Зв'язок "включення" на діаграмах варіантів використання показує:

- а) механізм успадкування властивостей класами-нащадками батьківського класу;

- б) процеси, що визначають і уточнюють цей процес;
- в) опис бізнес-процесів предметної області, що автоматизують.

53. Зв'язок "розширення" на діаграмах варіантів використання показує:

- а) збільшення кількості функцій бізнес-процесу;
- б) механізм успадкування властивостей класами-нащадками від батьківського класу;
- в) частину альтернативного потоку;
- г) визначення точки розширення для варіанту використання та її деталізацію (зміну функціональності) на основі цієї точки.

54. Основний потік відрізняється від альтернативного таким:

- а) наявністю різних передумов;
- б) наявністю умови його реалізації, на відміну від основного потоку;
- в) наявністю його виключення або включення.

7. Контрольні запитання для самодіагностики

Змістовий модуль 1

Основи тестування програмного забезпечення

Тема 1. Тестування як спосіб забезпечення якості програмного продукту

1. Назвіть шляхи подальшого розвитку методів та засобів тестування в Україні.
2. У чому полягає багаторівнева модель якості програмного забезпечення?
3. Охарактеризуйте поняття якості програмного забезпечення.
4. У чому полягають мета та завдання цієї навчальної дисципліни?
5. На знанні яких навчальних дисциплін базується опанування цієї навчальної дисципліни?
6. Вивчення яких навчальних дисциплін забезпечує ця навчальна дисципліна?
7. Наведіть показники якості програмного забезпечення.
8. Охарактеризуйте стандарт ISO.
9. Укажіть, хто із членів групи тестування відповідає за розроблення тестових специфікацій та аналіз результатів тестування.

10. Укажіть, хто із членів групи тестування відповідає за організацію процесу тестування.

11. Який із критеріїв тестування вказують в описі вимог до програмного продукту?

12. Наведіть визначення "якість програмного забезпечення", охарактеризуйте показники якості, способи контролю, стандарти.

13. Наведіть принципи формування команди розробників. Назвіть ролі учасників проекту.

14. Наведіть класифікацію вимог. Назвіть методи виявлення та документування вимог.

15. Назвіть способи опису вимог до програмного забезпечення.

16. Що таке "якість" із погляду кваліметрії?

17. Дати визначення властивості та показникам якості.

18. Назвіть основні завдання, які вирішують під час оцінювання якості програмного забезпечення.

Тема 2. Тестування на етапі розроблення програмного продукту

1. Назвіть функціональні фази тестування.

2. Охарактеризуйте особливості індустріального підходу в оцінюванні якості програмного продукту засобами тестування.

3. Наведіть класифікацію тестів за об'єктом тестування. Охарактеризуйте функціональне тестування та тестування продуктивності.

4. Охарактеризуйте тестування навантаження, стрес-тестування, тестування стабільності та конфігураційне тестування.

5. Назвіть склад групи тестування, охарактеризуйте права й обов'язки членів групи тестування.

6. Визначте місце тестування на етапах розроблення програмного продукту.

7. Назвіть критерії та методи системи менеджменту якості.

8. Визначте критерії вибору тестів.

9. Охарактеризуйте модульне та інтеграційне тестування.

10. На якому етапі тестування застосовують структурні критерії?

11. Яку умову застосовують під час проходження тестування кожної команди не менше від одного разу?

12. Які умови застосовують під час тестування процесу проходження кожного шляху?

13. Який із критеріїв тестування забезпечує контроль за ступенем виконання вимог замовника у програмному продукті?

14. Що таке "програмна система"?
15. Що таке "життєвий цикл програмного забезпечення"?
16. Перелічіть етапи життєвого циклу програмного забезпечення.
17. Опишіть основні підходи до програмування.
18. Опишіть структурний підхід до програмування.
19. Опишіть об'єктно-орієнтований підхід до програмування.
20. Охарактеризуйте роботу з кадрами. Визначте ролі розробників та дайте характеристику кожній із них.
21. Дати визначення проекту, процесу, продукту з погляду уніфікованого процесу розроблення програмного забезпечення.
22. Що таке "артефакт"?
23. У чому полягають переваги організованого процесу розроблення і тестування програмного забезпечення?

Тема 3. Тестування на етапах планування та проектування програмного продукту

1. У чому полягає структура тестового набору для автоматичного тестування?
2. У чому полягає структура інструментального середовища автоматизації тестування?
3. Назвіть типи тестів та їхню роль у процесі розроблення програмного забезпечення.
4. Визначте особливості тестування на етапі планування.
5. Визначте особливості тестування на етапі проектування.
6. У чому полягає автоматизація тестування?
7. Укажіть, хто із членів групи тестування відповідає за розроблення автоматизованих тестів, установлення і супроводження тестового оточення.
8. Яку модель тестування використовують під час функціонального тестування?
9. Яку модель тестування використовують під час модульного тестування?
10. Яку модель тестування використовують під час інтеграційного тестування?
11. До якого класу тестування можна зарахувати тестування інтерфейсів (екранів, кнопок тощо)?
12. Назвіть фактори, що визначають витрати на створення програмного забезпечення.

Тема 4. Тестування на стадії кодування

1. Обґрунтуйте вибір критеріїв якості тестування.
2. Охарактеризуйте статичне і динамічне тестування. У чому полягає їхня різниця?
3. Наведіть особливості тестування "скляної скриньки" на стадії кодування.
4. Дайте визначення регресійному тестуванню.
5. Наведіть особливості тестування "чорної скриньки".
6. Як здійснюють забезпечення якості програмного продукту засобами тестування?
7. Який із критеріїв тестування використовує інформацію про структуру програми?
8. Дайте визначення тестування та налагодження.
9. Наведіть особливості й об'єкти тестування.
10. Визначте особливості автономного та комплексного тестування.
11. Назвіть напрями тестування.
12. У чому полягає стратегія тестування?

Тема 5. Програмні помилки

1. Охарактеризуйте життєвий цикл помилки.
2. Назвіть особливості застосування різних критеріїв оцінювання швидкості виявлення помилок.
3. Поясніть поняття програмної помилки.
4. Визначте категорії програмних помилок.
5. Наведіть приклади помилки інтерфейсу користувача.
6. Як здійснюють оброблення помилок?
7. Визначте помилки, пов'язані з обробленням граничних умов.
8. Визначте помилки передавання даних.
9. Визначте помилки оброблення даних.
10. У чому полягає перенавантаження?
11. Укажіть, хто із членів групи тестування відповідає за виконання тестів і реєстрацію помилок.
12. До якого класу тестів належить перевірка правильності роботи системи?
13. До якого класу тестів зараховують перевірку правильності оброблення даних?

14. Наведіть класифікацію помилок.
15. У чому полягає локалізація помилок?
16. У чому полягає оцінювання помилок?
17. У чому полягає безпечне програмування?

Змістовий модуль 2

Організація процесу тестування програмного забезпечення

Тема 6. Система відстеження дефектів. Планування тестування

1. У чому полягає особливість тестування на основі специфікацій та сценаріїв?
2. Охарактеризуйте регресійне тестування.
3. Охарактеризуйте альфа-тестування.
4. Охарактеризуйте бета-тестування.
5. У чому полягає сутність релізу?
6. У чому полягає процес документування тестових процедур для ручних та автоматизованих тестів?
7. Назвіть основне призначення систем відстеження дефектів.
8. Визначте різновиди систем багтрекінгу.
9. Наведіть структуру звіту про проблему.
10. Охарактеризуйте класи еквівалентності та граничні умови.
11. У чому полягає тестування функціональності?
12. У чому полягає тестування переходів між станами?
13. У чому полягає тестування випробування навантажень?

Тема 7. Документування тестових матеріалів

1. Охарактеризуйте регресійне тестування.
2. У чому полягає розроблення тестових сценаріїв та процедур?
3. У чому полягає виконання тестів?
4. Назвіть компоненти плану тестування.
5. Визначте загальне призначення тестового плану.
6. Охарактеризуйте процес документування тестових матеріалів.
7. Назвіть види звітів про виконання тестування.
8. Які документи використовують для розроблення плану тестування?
9. Які документи використовують для розроблення тестових специфікацій?

10. Який документ призначено для документування результатів виконання плану тестування?
11. Які існують документи щодо процесу тестування програмних продуктів?
12. Які процеси під час тестування не підлягають документуванню?

Тема 8. Автоматизація тестування

1. Дайте визначення автоматизації тестування.
2. Визначте завдання автоматизації тестування.
3. Дайте визначення комбінації методів тестування.
4. Наведіть етапи виконання тестів.
5. У чому полягає ручне тестування?
6. У чому полягає автоматизоване тестування?
7. Зробіть порівняння ручного й автоматизованого тестування.

Тема 9. Особливості індустріального тестування

1. У чому полягає індустріальний підхід?
2. Наведіть особливості індустріального тестування.
3. Наведіть визначення якості програмного продукту та тестування.
4. У чому полягає оцінювання індустріального тестування?
5. Наведіть особливості документування індустріального тестування.
6. Що таке "промисловий програмний продукт"?
7. Дайте визначення пакета прикладних програм програмної системи.

Тема 10. Проблеми якості програмного забезпечення та практичні рекомендації

1. У чому полягає наскрізне забезпечення якості програмного забезпечення?
2. Назвіть вимоги до якості програмного забезпечення.
3. У чому полягає виконання тестів?
4. У чому полягає тестування документації?
5. Назвіть стандарти розроблення програмного забезпечення та їхнього використання.
6. Охарактеризуйте методики забезпечення якості програмного забезпечення в межах усього життєвого циклу розроблення.
7. Назвіть склад технологічної документації на програмні засоби.

8. Охарактеризуйте експлуатаційну документація на програмні засоби.
9. У чому полягає сесійне тестування?
10. У чому полягає тестування, засноване на ризиках?
11. Наведіть структуру та зміст документів за етапами життєвого циклу програмних засобів.
12. Назвіть склад документації користувача на програмні засоби.
13. Охарактеризуйте технічне завдання на проектування програмних засобів.
14. Охарактеризуйте ескізний і робочий проект програмних засобів.
15. Охарактеризуйте документацію тестування компонентів і комплексів програм.
16. Назвіть склад документації супроводу та конфігураційного управління версіями програм.

8. Рекомендована література

8.1. Основна

1. Бейзер Б. Тестирование чёрного ящика. Технология функционального тестирования / Б. Бейзер. – Санкт-Петербург, 2004. – 221 с.
2. Бек К. Экстремальное программирование: планирование / К. Бек. – Санкт-Петербург : Питер, 2003. – 143 с.
3. Богданов Д. В. Стандартизация процессов обеспечения качества программного обеспечения / Д. В. Богданов, В. А. Путилов, В. В. Фильчаков. – Апатиты : КФ ПетрГУ, 1988. – 152 с.
4. Брауде Э. Технология разработки программного обеспечения / Э. Брауде. – Санкт-Петербург : Питер, 2004. – 410 с.
5. Гриняк В. М. Использование методов программной инженерии в процессе обучения дисциплинам типа "Программирование" / В. М. Гриняк, Н. Л. Слугина // Современные проблемы науки и образования, 2013. – № 2. – С. 56–61.
6. Дастин Э. Автоматизированное тестирование программного обеспечения / Э. Дастин, Дж. Рэшка, Дж. Пол. – Москва : Изд-во "ЛОРИ", 2003. – 567 с.
7. Калбертсон Р. Быстрое тестирование / Р. Калбертсон, К. Браун, Г. Кобб. – Москва : Вильямс, 2002. – 374 с.

8. Канер С. Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений / С. Канер, Дж. Фолк, Е. Нгуен. – Киев : ДиаСофт, 2001. – 544 с.
9. Ковальов А. И. Управление проектом по созданию интернет-сайта / А. И. Ковальов. – Москва : Альпина, 2004. – 338 с.
10. Котляров В. П. Основы тестирования программного обеспечения : учеб. пособ. / В. П. Котляров, Т. В. Коликова. – Москва : БИНОМ. Лаборатория знаний, Интернет-университет информационных технологий, 2006. – 285 с.
11. Криспин Л. Гибкое тестирование: практическое руководство для тестировщиков ПО и гибких команд / Л. Криспин, Дж. Грегори. – Москва : Вильямс, 2010. – 464 с.
12. Липаев В. В. Выбор и оценивание характеристик качества программных средств: Методы и стандарты / В. В. Липаев ; Моск. гос. техн. ин-т "Станкин". – Москва : Синтег, 2001. – 224 с.
13. Липаев В. В. Качество программных средств / В. В. Липаев. – Москва : Янус-К, 2012. – 399 с.
14. Липаев В. В. Программная инженерия. Методологические основы / В. В. Липаев. – Москва : ТЕИ, 2006. – 608 с.
15. Липаев В. В. Сертификация программных средств : учебник / В. В. Липаев. – Москва : СИНТЕГ, 2010. – 344 с.
16. Липаев В. В. Тестирование программ / В. В. Липаев. – Москва : Радио и связь, 1986. – 296 с.
17. Липаев В. В. Управление разработкой ПС: методы, стандарты, технология / В. В. Липаев. – Москва : Финансы и статистика, 1993. – 160 с.
18. Майерс Г. Искусство тестирования программ / Г. Майерс. – Москва : Финансы и статистика, 1982. – 176 с.
19. Макгрегор Дж. Тестирование объектно-ориентированного программного обеспечения / Дж. Макгрегор, Д. Сайкс. – Киев : ООО "ТИД ДС", 2002. – 432 с.
20. Основы инженерии качества программных систем / Ф. И. Андон, Г. И. Коваль Т. М., Коротун и др. ; НАН Украины, Ин-т прогр. систем. – Киев : ИД "Академпериодика", 2002. – 502 с.
21. Савин Р. Тестирование Дот Ком или Пособие по жестокому обращению с багами в интернет-стартапах / Р. Савин. – Москва : Дело, 2007. – 312 с.

22. Синицын С. В. Верификация программного обеспечения / С. В. Синицын, Н. Ю. Налютин. – Москва : БИНОМ, 2008. – 368 с.

23. Слугина Н. Л. Повышение уровня подготовки кадров в области современных информационных технологий на базе центра компетенций / Н. Л. Слугина, В. М. Гриняк // Современные проблемы науки и образования. – 2012. – № 5. – С. 205–215.

24. Соммервилл И. Инженерия программного обеспечения / И. Соммервилл. – 6-е изд. – Москва : Вильямс, 2002. – 642 с.

25. Тамре Л. Введение в тестирование программного обеспечения / Л. Тамре. – Москва : Вильямс, 2003. – 368 с.

26. Уиттакер Дж. Как тестирует Google / Дж. Уиттакер, Дж. Арбон, Дж. Каролло. – Санкт-Петербург : Питер, 2014. – 320 с.

8.2. Додаткова

27. Боэм Б. Характеристики качества программного обеспечения / Б. Боэм, Дж. Браун, Х. Каспар. – Москва : Мир, 1981. – 208 с.

28. Бураков В. В. Методика оценки качества программных средств / В. В. Бураков // Известия вузов. Приборостроение. – 2008. – Т. 51, № 1. – С. 13–18.

29. Гацко Г. Тестирование ПО как один из элементов системы качества / Г. Гацко // Открытые системы. – № 06. – 1998. – С. 125–128.

30. Гласс Р. Сопровождение программного обеспечения / Р. Глас ; под ред. Ю. А. Чернышева. – Москва : Мир, 1983. – 158 с.

31. Долгов Ю. А. Количественная оценка некоторых характеристик надежности программного обеспечения / Ю. А. Долгов, Т. Г. Данилина // Радіоелектронні і комп'ютерні системи. – 2006. – № 7 (19). – С. 152–155.

32. Долгов Ю. А. Статистическое моделирование : учеб. для вузов / Ю. А. Долгов. – 2-е изд., доп. – Тирасполь : Полиграфист, 2011. – 352 с.

33. Домаратский А. Н. Управление улучшением стандартного процесса и качеством программных изделий / А. Н. Домаратский // Программные продукты и системы. – 1998. – № 4. – С. 20–24.

34. Иванова Г. С. Технология программирования : учеб. для вузов / Г. С. Иванова. – Москва : МГТУ им. Н. Э. Баумана, 2002. – 320 с.

35. Канер С. Тестирование программного обеспечения / С. Канер. – Киев : ДиаСофт, 2000. – 318 с.

36. Калинина Л. Ю. Оценка качества программных продуктов / Л. Ю. Калинина // Качество. Инновации. Образование. – 2006. – № 4. – С. 52–55.
37. Кантор М. Управление программными проектами. Практическое руководство по разработке успешного программного обеспечения / М. Кантор. – Москва : Вильямс, 2002. – 176 с.
38. Коваленко И. И. Оценка качества программных продуктов с использованием теории Демпстера – Шейфера / И. И. Коваленко, А. В. Швед // Наукові праці ЧДУ. Комп'ютерні технології. – 2011. – Т. 160, № 148. – С. 22–26.
39. Когаловский М. Р. Энциклопедия технологий баз данных / М. Р. Когаловский. – Москва : Финансы и статистика, 2005. – 800 с.
40. Коул Дж. Принципы тестирования ПО / Дж. Коул, Т. Горэм // Открытые системы. – 1998. – № 2. – С. 34–37.
41. Кречетов Н. Технологии оценки качества программных продуктов / Н. Кречетов // Computer Week. – 1996. – № 25. – Р. 27–32.
42. Майерс Г. Искусство тестирования программ / Г. Майерс, Т. Баджетт, К. Сандлер. – 3-е изд. – Москва : Диалектика, 2012. – 272 с.
43. Менчер Э. М. Обобщенная функция полезности / Э. М. Менчер // Радионуклеиды и тонирующие излучения в исследованиях по виноградарству. – Кишинев : Штиинца, 1983. – С. 104–118.
44. Рэшка Дж. Тестирование программного обеспечения / Дж. Рэшка, Э. Дастин, Дж. Пол. – Москва : Лори, 2012. – 568 с.
45. Скорін Ю. І. Методика кількісної оцінки ефективності роботи інженера із забезпечення якості програмного забезпечення / Ю. І. Скорін, О. В. Щербаков, Є. С. Луценко // Системи обробки інформації : збірник наукових праць. – У 2-х т. Т1. Інформаційні технології та захист інформації. – Харків : ХУПС ім. І. Кожедуба. – 2012. – Вип. 4 (102). – С. 81–84.
46. Скорин Ю. И. Модуль баг-трекинга при разработке программного обеспечения / Ю. И. Скорин, А. А. Пивень, В. В. Стадник // Системи обробки інформації : збірник наукових праць III Міжнародної науково-практичної конференції [Проблеми і перспективи розвитку ІТ-індустрії]. (Харків, 17 – 18 листопада 2011 року) – Харків : ХУПС ім. І. Кожедуба. – 2011. – Вип. 7(97). – С. 111.
47. Скорин Ю. И. Тестирование программного обеспечения / Ю. И. Скорин, А. А. Пивень // Системи обробки інформації : збірник наукових праць. – У 2-х т. Т1. Інформаційні технології та захист інформації. Харків : ХУПС ім. І. Кожедуба. – 2012. – Вип. 4 (102). – С. 56–58.

48. Стивенс Р. Тестирование и отладка программ на Visual Basic / Р. Стивенс. – Москва : Прогресс, 2001. – 384 с. Серия "Для программистов".
49. Трофимов С. А. CASE-технологии : практическая работа в Rational Rose / С. А. Трофимов. – Москва : Бином-Пресс, 2002. – 288 с.
50. Уилсон С. Принципы проектирования и разработки программного обеспечения. Учебный курс MCSD / С. Уилсон, Б. Мэйлс, Т. Ленгрейв. – Москва : Русская редакция, 2000. – 608 с.
51. Фаулер М. UML. Основы / М. Фаулер. – Санкт-Петербург : Символ-Плюс, 2005. – 192 с.
52. Фаулер М. Рефакторинг: улучшение соответствующего кода / М. Фаулер. – Санкт-Петербург : Символ-Плюс, 2003. – 432 с.
53. Фаулер М. UML в кратком изложении. Применение стандартного языка объектного моделирования / М. Фаулер, К. Скотт. – Москва : Мир, 1999. – 191 с.
54. Холл М. Сервлеты и JavaServer Pages. Библиотека программиста / М. Холл. – Санкт-Петербург : Питер, 2001. – 496 с.
55. Чернецки К. Порождающее программирование. Методы, инструменты, применение / К. Чернецки, У. Айзенекер. – Санкт-Петербург : Питер, 2005. – 730 с.
56. Шафер Д. Управление программными проектами: достижение оптимального качества при минимуме затрат / Д. Шафер, Р. Фатрел, Л. Шафер. – Москва : Вильямс, 2003. – 136 с.
57. Шилдт Г. Программирование на С и С++ для Windows 95 / Г. Шилдт. – Киев : BHV, 1996. – 400 с.
58. Шилдт Г. Теория и практика С++ / Г. Шилдт. – Санкт-Петербург : BHV, 1999. – 416 с.
59. Шилдт Г. МFC: основы программирования / Г. Шилдт. – Киев : BHV, 1997. – 560 с.
60. Шлеер С. Объектно-ориентированный анализ: моделирование мира в состояниях / С. Шлеер, С. Меллор. – Киев : Диалектика, 1993. – 240 с.
61. Шмуллер Дж. Освой самостоятельно UML за 24 часа / Дж. Шмуллер. – Москва : Вильямс, 2005. – 135 с.
62. Эммерих В. Конструирование распределенных объектов. Методы и средства программирования интероперабельных объектов в архитектурах OMG/CORBA, Microsoft COM и Java RMI / В. Эммерих. – Москва : Мир, 2002. – 510 с.

8.3. Інформаційні ресурси

63. Библиотека MSDN. Источник информации для разработчиков, использующих средства, продукты, технологии и службы корпорации Майкрософт [Электронный ресурс]. – Режим доступа : <http://msdn.-microsoft.com>.

64. ИНТУИТ. Интернет-университет информационных технологий [Электронный ресурс]. – Режим доступа : <http://intuit.ru>.

65. Качество программного обеспечения [Электронный ресурс]. – Режим доступа : <http://openquality.ru/main>.

66. КиберФорум. Форум начинающих и профессиональных программистов, системных администраторов, администраторов баз данных [Электронный ресурс]. – Режим доступа : <http://cyberforum.ru>.

67. Клуб программистов [Электронный ресурс]. – Режим доступа : <http://www.cyberguru.ru>.

68. Липаев В. В. Документирование сложных программных средств [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://www.computer-museum.ru/books/lipaev>.

69. Липаев В. В. Надежность и функциональная безопасность комплексов программ реального времени [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://www.computer-museum.ru/books/lipaev>.

70. Липаев В. В. Программная инженерия в жизненном цикле программных средств [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://citforum.ru/SE/lipaev>.

71. Липаев В. В. Программная инженерия. Методологические основы. Предисловие к курсу [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://citforum.ru/SE/lipaev/preface.shtml>.

72. Липаев В. В. Проектирование и производство сложных заказных программных продуктов [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://www.computer-museum.ru/books/lipaev>.

73. Липаев В. В. Сопровождение и управление конфигурацией сложных программных средств [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://www.computer-museum.ru/books/lipaev>.

74. Липаев В. В. Стандарты, регламентирующие жизненный цикл сложных программных комплексов [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://www.pcweek.ru/year1998/N24/CP1251/Reviews/chapt1.htm>.

75. Липаев В. В. Тестирование компонентов и комплексов программ [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://www.computer-museum.ru/books/lipaev>.

76. Липаев В. В. Экономика производства программных продуктов [Электронный ресурс] / В. В. Липаев. – Режим доступа : <http://www.computer-museum.ru/books/lipaev>.

77. Марченко Е. Что такое качество программного обеспечения? [Электронный ресурс] / Е. Марченко. – Режим доступа : <http://www.silicontaiga.ru/home.asp?artId=2862>.

78. Портал об автоматизированном тестировании ПО. – Режим доступа : <http://automated-testing.info>.

79. Портал специалистов по тестированию и обеспечению качества ПО. – Режим доступа : <http://software-testing.ru>.

80. Про тестинг [Электронный ресурс]. – Режим доступа : <http://www.protesting.ru>.

81. Самостоятельная работа студентов [Электронный ресурс]. – Режим доступа : <http://www.studfiles.ru/preview/3495098>.

82. Силласте Г. Г. Самостоятельная работа студентов: методические рекомендации [Электронный ресурс] / Г. Г. Силласте. – Режим доступа : https://docviewer.yandex.ua/?url=http%3A%2F%2F-fmggush.ru%2Fsites%2Fdefault%2Ffiles%2Fmetod.rekom_sam.rab_dlya_studentov.pdf&name=metod.rekom_sam.rab_dlya_studentov.pdf&lang=ru&c=583593fd0dce&page=3.

83. Тестирование и качество ПО [Электронный ресурс]. – Режим доступа : <http://software-testing.ru>.

84. Тестирование программного обеспечения [Электронный ресурс]. – Режим доступа : [https://ru.wikipedia.org/wiki/test%](https://ru.wikipedia.org/wiki/test%2F).

8.4. Методичне забезпечення

85. Робоча програма навчальної дисципліни "Технології створення програмних продуктів" для студентів напряму підготовки 6.050101 "Комп'ютерні науки" денної форми навчання / уклад. Ю. І. Скорін. – Харків : ХНЕУ, 2016. – 73 с.

86. Робоча програма навчальної дисципліни "Технології тестування програмних продуктів" для студентів за напрямом підготовки 6.050101 "Комп'ютерні науки" усіх форм навчання / уклад. Ю. І. Скорін. – Харків : Вид. ХНЕУ ім. С. Кузнеця, 2012. – 48 с.

Зміст

Вступ.....	3
1. Самостійна робота студентів	8
2. Методичні рекомендації щодо підготовки до лекційних занять, роботи з рекомендованою літературою.....	15
3. Методичні рекомендації щодо підготовки до виконання практичних завдань	17
4. Перелік тем есе.....	18
5. Методичні рекомендації щодо підготовки до поточного тестового контролю (контрольної роботи)	20
6. Перелік запитань тесту.....	25
7. Контрольні запитання для самодіагностики	34
8. Рекомендована література.....	40
8.1. Основна	40
8.2. Додаткова	42
8.3. Інформаційні ресурси.....	45
8.4. Методичне забезпечення	46

НАВЧАЛЬНЕ ВИДАННЯ

ЯКІСТЬ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕСТУВАННЯ

**Методичні рекомендації
до самостійної роботи студентів
спеціальності 121 "Інженерія програмного забезпечення"
першого (бакалаврського) рівня**

Самостійне електронне текстове мережеве видання

Укладач **Скорін Юрій Іванович**

Відповідальний за видання *О. Г. Руденко*

Редактор *О. Г. Доценко*

Коректор *О. Г. Доценко*

План 2019 р. Поз. № 164 ЕВ. Обсяг 48 с.

Видавець і виготовлювач – ХНЕУ ім. С. Кузнеця, 61166, м. Харків, просп. Науки, 9-А

*Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру
ДК № 4853 від 20.02.2015 р.*