**Лекція №25. Моделі якості і надійності програмних систем**

Основою інженерних методів у програмуванні є підвищення якості. Для досягнення цього були  сформульовані  методи визначення вимог до якості, підходи до вибору й удосконалення моделей метричного аналізу показників якості, методи кількісного виміру ризиків  на процесах життєвого циклу (далі – ЖЦ).

Головна складова якості –**надійність**, якій приділяється велика увага у сфері надійності технічних засобів і тих критичних систем (реальний час, радарні системи, системи безпеки й ін.), для яких надійність є головною цільовою функцією оцінки їхньої реалізації. Як наслідок у проблематиці надійності розроблено понад сотні математичних моделей надійності, що є функціями  помилок, які залишилися в програмній системі (далі – ПС),  інтенсивності відмов або частоти виникнення дефектів у ПС. На їхній основі здійснюється оцінка надійності ПС.

Якість ПС – предмет стандартизації. У стандарті ДСТУ 2844–94 наведено визначення якості ПС як сукупності властивостей (показників якості) ПС, *що забезпечують його здатність задовольняти потреби замовника відповідно до призначення*. Цей стандарт регламентує базову модель якості й показники, головним серед яких є надійність. Стандарт ISO/IEC 12207 визначає не тільки основні процеси ЖЦ розробки ПС, а й організаційні та додаткові процеси, які регламентують інженерію, планування й керування якістю ПС.

Відповідно до стандарту на процесах ЖЦ повинен здійснюватися контроль якості ПС:

- перевірка відповідності вимог до проектованого продукту та критеріїв їхнього досягнення;

- верифікація й атестація (валідація) проміжних результатів ПС на процесах ЖЦ і вимірювання ступеня відповідно до  певних показників, які досягаються;

- тестування готової ПС, збирання даних про відмови, дефекти й інші помилки, які виявлено у системі;

- підбір моделей надійності для оцінювання надійності за отриманими результатами тестування (дефекти, відмови й ін.);

- оцінка показників якості, заданих у вимогах до  розроблення ПС.

**Модель якості програмних систем**

Якість ПС – це відносне поняття, що має сенс тільки з урахуванням реальних умов його застосування, тому вимоги до якості висуваються відповідно до умов та конкретної сфери їхнього використання. Якість характеризується трьома аспектами: якість програмного продукту, якість процесів ЖЦ й якість супроводу або впровадження (рис. 1).

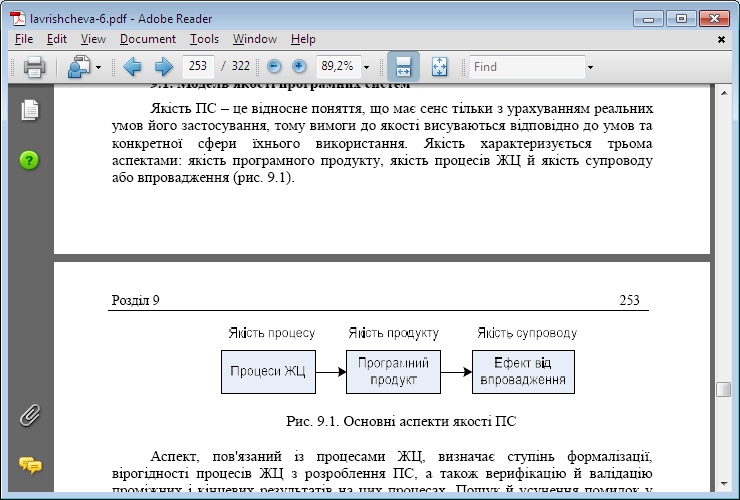


Рис. 1. Основні аспекти якості ПС

Аспект, пов'язаний із процесами ЖЦ, визначає ступінь формалізації, вірогідності процесів ЖЦ з розроблення ПС, а також верифікацію й валідацію проміжних і кінцевих результатів на цих процесах. Пошук й усунення помилок у готовому ПС проводиться за допомогою методів тестування, які зменшують кількість помилок і підвищують якість цього продукту.

Якість продукту досягається процедурами контролю проміжних продуктів під час  процесів ЖЦ, перевіркою їх на досягнення необхідної якості, а також методами супроводу продукту. Ефект від впровадження ПС великою мірою залежить від знань обслуговуючого персоналу функцій продукту й правил їхнього виконання.

Модель якості програмного забезпечення (яку ми також розглядаємо в курсі конструювання ПЗ) має чотири рівні подання.

**Перший рівень**подання відповідає визначенню характеристик (показників) якості ПС, кожна з яких відображає окреме уявлення користувача про якість.

Відповідно до стандарту у модель якості входить шість характеристик або шість показників якості:

1) функціональність (functionality);

2) надійність (realibility);

3) зручність (usability);

4) ефективність (efficiency);

5) супровід (maitainnability);

6) мобільність (portability).

**Другому рівню**подання відповідають атрибути для кожної характеристики якості, які деталізують різні аспекти конкретної характеристики. Набір атрибутів характеристик якості використовується для оцінки якості.

**Третій рівень**подання призначено для виміру якості за допомогою метрик, кожна з яких відповідно до стандарту визначається як комбінація методу виміру атрибута й шкали виміру значень атрибутів. Для оцінки атрибутів якості на процесах ЖЦ (при перегляді документації, програм і результатів тестування програм) використовуються метрики із заданою цінною вагою для нівелювання результатів метричного аналізу сукупних атрибутів конкретного показника і якості в цілому. Атрибут якості визначається за допомогою однієї або декількох методик оцінки на процесах ЖЦ і на завершальному процесі розроблення ПС.

**Четвертий рівень**подання – це оцінний елемент метрики (вага), що використовується для оцінки кількісного або якісного значення окремого атрибута показника ПС. Залежно від призначення, особливостей та умов супроводу  вибираються найважливіші характеристики якості та їхні атрибути.

# Стандартні показники якості

Короткий опис семантики характеристик моделі якості ми розгляднемо в курсі конструювання програмного забезпечення, їх наведено в табл. 1, нижче розглянемо їх змістовний опис.

Таблиця 1**–**Коротка характеристика показників якості

| **Показник** | **Опис властивостей показника** |
| --- | --- |
| Функціональність | Група властивостей ПС, що обумовлює його здатність виконувати певний перелік функцій, які задовольняють потреби відповідно до призначення |
| Надійність | Група властивостей, що обумовлює здатність ПС зберігати працездатність і перетворювати вихідні дані на результат за встановлений період часу, характер відмов якого є наслідком внутрішніх дефектів й умов його застосування |
| Зручність  застосування | Сукупність властивостей ПС для передбачуваного кола користувачів й освоєння, що характеризують його простоту і адаптації до умов, що змінюються, експлуатації, стабільність роботи й підготовки даних, зрозумілість результатів, зручності внесення змін у програмну документацію й  програми |
| Супровід | Група властивостей, що визначає зусилля, необхідні для виконання, пристосовність до діагностики відмов і наслідків внесення змін, модифікації й атестації ПС, що модифікується |
| Ефективність/ Раціональність | Група властивостей, що характеризується ступенем відповідності використовуваних ресурсів середовища до функціонування рівня якості (надійності),  функціонування ПС за заданих умов застосування |
| Переносність | Група властивостей ПС, що забезпечує його пристосовність для перенесення з одного середовища функціонування в інші, зусилля для перенесення й адаптації ПС до нового середовища функціонування |

**1. Функціональність** – сукупність властивостей, що визначають здатність ПС виконувати перелік функцій у певному середовищі відповідно до вимог щодо обробки й загальносистемних засобів.

*Функція* розглядається як певна впорядкована послідовність дій для задоволення споживацьких нахилів. Функції бувають цільові (основні) і допоміжні.

До атрибутів функціональності належать:

- функціональна повнота – властивість компонента, що визначає ступінь достатності основних функцій для розв’язання задач відповідно до призначення ПС;

- правильність (точність) – атрибут, що визначає ступінь досягнення правильних результатів;

- інтероперабельність – атрибут, що визначає можливість впливати на ПС спеціальними системами й середовищами (ОС, мережа);

- захищеність – атрибут, що вказує на здатність ПС запобігати несанкціонованому доступу (випадковий або навмисний) до програм і даних.

**2. Надійність** – сукупність атрибутів, які визначають здатність ПС перетворювати вихідні дані на результати за умов, що залежать від періоду часу життя (зношування і старіння не враховуються). Зниження надійності ПС відбувається через помилки у вимогах, проектуванні й виконанні. Відмови й помилки в програмах з'являються на визначеному проміжку часу.

До атрибутів  надійності ПС належать:

- безвідмовність – атрибут, що визначає здатність ПС функціонувати без відмов (як програми, так й обладнання);

- стійкість до помилок – атрибут, що вказує на здатність ПС виконувати функції за аномальних умов (збій апаратури, помилки в даних й інтерфейсах, порушення в діях оператора й ін.);

- відновлюваність – атрибут, що вказує на здатність програми до перезапуску для повторного виконання й відновлення даних після відмов.

До певних типів систем (реального часу, радарних, систем безпеки, комунікація й ін.) висуваються вимоги щодо забезпечення високої надійності (неприпустимість помилок, точність, вірогідність, зручність застосування й ін.). Таким чином, надійність ПС значною мірою залежить від кількості не усунутих помилок у процесі розроблення продукту на процесах ЖЦ. Під час експлуатації помилки виявляються й усуваються.

Якщо при виправленні помилок не вносяться нові або, принаймні, нових помилок вноситься менше, ніж усувається, то і в процесі експлуатації надійність ПС зростає. Чим інтенсивніший процес експлуатації, тим інтенсивніше виявляються помилки й швидше зростає надійність ПС.

До чинників, що впливають на надійність ПС, належать:

- сукупність загроз, що призводять до несприятливих наслідків і збитків системи або середовища її функціонування;

- загроза як виявлення порушення безпеки системи;

- цілісність як здатність системи зберігати стійкість роботи без  ризику.

Виявлені помилки можуть бути результатом загрози ззовні або відмов, вони підвищують ризик і зменшують деякі властивості надійності системи.

Надійність – одна з ключових проблем сучасних програмних систем, і її роль буде постійно зростати, оскільки постійно підвищуються вимоги до якості комп'ютерних систем. Новий напрямок – інженерія програмної надійності (Software reliability engineering) – орієнтується на кількісне вивчення операційної поведінки компонентів системи відносно  користувача, який очікує надійну роботу системи. Він складається з:

- вимірювання надійності, тобто проведення її кількісної оцінки за допомогою передбачень, збирання даних про поведінку системи в процесі експлуатації й сучасних моделей надійності;

- стратегії й метрики конструювання й вибору готових компонентів, процес розробки компонентної системи, а також середовище функціонування, що впливає на надійність роботи системи;

- застосування сучасних методів інспектування, верифікації, валідації й тестування під час розроблення систем та їх експлуатації.

Верифікація застосовується для встановлення відповідності готового ПС до визначених специфікацій, а валідація – для встановлення відповідності системи до вимог користувача, висунутих замовником.

Оцінка надійності ПС – це трудомісткий процес, що вимагає створення сталої роботи системи щодо помилок  ПС, тобто ймовірності того, що система відновиться довільно в певній точці після виникнення і внесення змін через  відмови і інші ситуації.

**3. Зручність застосування** характеризується множиною атрибутів, які вказують на необхідні й достатні умови використання (діалогове або не діалогове) ПС певним колом користувачів для отримання відповідних результатів. У стандарті визначено зручність застосування  як специфічну множину атрибутів програмного продукту, що характеризують його ергономічність.

До атрибутів зручності застосування належать:

- зрозумілість – атрибут, що визначає зусилля, витрачені на розпізнавання логічних концепцій та умов застосування ПС;

- легкість вивчення – атрибут, що визначає зусилля користувачів, спрямовані на визначення застосування ПС шляхом використання операційного контролю, діагностики, а також процедур, правил і документації;

- оперативність – атрибут, що вказує на реакцію системи під час виконання операцій й операційного контролю;

- погодженість – атрибут, що визначає відповідність розроблення вимогам стандартів, угод, правил, законів і розпоряджень.

**4. Ефективність** – множина атрибутів, які визначають взаємозв'язок рівнів виконання ПС, використання ресурсів (засоби, апаратура, матеріали – папір для друкувального пристрою та ін.) і послуг, що виконуються штатним обслуговуючим персоналом та ін.

До характеристик ефективності ПС належать:

- реактивність – атрибут, що вказує на час відгуку, обробки й виконання функцій;

- ефективність ресурсів – атрибут, що визначає кількість і тривалість використовуваних ресурсів при виконанні функцій ПС;

- погодженість – атрибут, що вказує на відповідність даного атрибута заданим стандартам, правилам та приписам.

**5. Супровід** – множина властивостей, які вказують на зусилля, що треба витратити для здійснення модифікацій, що містить у собі корегування, удосконалення й адаптацію ПС при зміні середовища, вимог або функціональних специфікацій.

Супровід містить у собі такі атрибути:

- простота аналізу – атрибут, що визначає необхідні зусилля для діагностики відмов або ідентифікації частин, які будуть модифікуватися;

- змінюваність – атрибут, що визначає видалення помилок у ПС або внесення змін для їхнього усунення, а також введення нових можливостей до ПС або до середовища функціонування;

- стабільність – атрибут, що вказує на сталість структури й ризик її модифікації;

- простота тестування – атрибут, що вказує на зусилля при проведенні валідації й верифікації з метою виявлення невідповідностей вимогам, а також на необхідність проведення модифікації ПС й сертифікації;

- погодженість – атрибут, що показує відповідність даного атрибута угодам, правилам і приписам стандарту.

**6. Переносність** – множина показників, що вказують на здатність ПС адаптуватися до роботи в нових умовах середовища виконання. Середовище може бути організаційне, апаратне й програмне. Тому перенесення ПС в нове середовище виконання може бути пов'язано з сукупністю дій, спрямованих на забезпечення його функціонування в середовищі, відмінному від того середовища, у якому воно створювалося з урахуванням нових програмних, організаційних і технічних можливостей.

Переносність містить у собі такі  атрибути:

- адаптивність  визначає зусилля, що витрачаються на адаптацію до різних середовищ;

- простота налагодження  визначає необхідні зусилля для запуску певного ПС в спеціальному середовищі;

- співіснування визначає можливість використання спеціального ПС в середовищі діючої системи;

- замінність забезпечує можливість інтероперабельності при спільній роботі з іншими програмами з необхідною інсталяцією або адаптацією ПС;

- узгодженість – це відповідність стандартам або угодами про забезпечення перенесення ПС в інше середовище.

**Моделі надійності**

 В останні роки вартість розроблення програмного забезпечення (далі - ПЗ) та витрати на відмови програмних систем стали одними із основних затрат на розроблення складних технічних систем. Відмова ПЗ може призвести до неочікуваного стану системи або потоку непередбачених дій. Внаслідок відмови ПЗ можливі пошкодження чи руйнування майна, поранення або загибель людей, втрата грошових коштів.

Загальновизнаним в індустрії програмного забезпечення є той факт, що висококваліфіковані програмісти допускають в середньому 6 помилок в програмному забезпеченні при написанні 1000 рядків коду. При такому показнику звичайне комерційне ПЗ, яке складається з 350000 рядків коду, містить близько 2000 помилок, зокрема помилки, пов’язані з витоком пам’яті, пов’язані з текстом програми, помилки використання сторонніх бібліотек, стандартні бібліотечні помилки та інші. Тому питання оцінювання та забезпечення надійності ПЗ відіграють все більшу роль у розробленні складних технічних систем, а побудова все адекватніших моделей надійності ПЗ є важливою науково-технічною проблемою.

**Загальна характеристика та класифікація моделей надійності програмного забезпечення**

Дослідження в галузі надійності ПЗ беруть початок з 1970-х років. Незважаючи на зусилля провідних світових розробників ПЗ, завдання зниження кількості помилок у програмних системах не отримало ефективного практичного вирішення. Об’єктивно це зумовлено надзвичайно високою структурною складністю програмних систем, динамічністю версій і технологій. Одним із шляхів підвищення рівня надійності ПЗ є використання на етапах тестування і випробувань ПЗ моделей, що дають змогу отримати гарантовані оцінки показників безпеки ПЗ і ефективності технології його розроблення. Більшість таких моделей запозичено з теорії надійності технічних систем, тому в літературі їх часто називають моделями надійності ПЗ.

Надійність складних ПС істотно відрізняється від надійності апаратури. Носії даних (файли, сервер тощо) мають високу надійність, записи на них можуть зберігатися тривалий час без руйнування, оскільки фізичному руйнуванню вони практично не підлягають.

З погляду прикладної науки *надійність* – це здатність ПС зберігати свої властивості (безвідмовність, стійкість та ін.), перетворювати вихідні дані на результати протягом певного проміжку часу за певних умов експлуатації. Зниження надійності ПС відбувається через помилки у вимогах, проектуванні й виконанні. Відмови й помилки залежать від способу виробництва продукту й з'являються в програмах при їхньому виконанні на деякому проміжку часу.

Таким чином, оцінка надійності ПС залежить від числа помилок, що залишилися не усунутими у програмах. У процесі експлуатації ПС помилки виявляють й усувають. Якщо при виправленні помилок не вносяться нові або, принаймні, нових помилок вноситься менше, ніж усувається, то в процесі експлуатації надійність ПС безперервно зростає. Чим інтенсивніше здійснюється експлуатація, тим інтенсивніше виявляються помилки й швидше зростає надійність системи й відповідно її якість.

Надійність є функцією  помилок, що залишилися в ПС після введення його до експлуатації. ПС без помилок є абсолютно надійним. Але для більшості програм абсолютна надійність практично недосяжна. Помилки, що залишилися незнайденими, виявляють себе час від часу за певних умов (наприклад, при певній сукупності вихідних даних) супроводу й експлуатації системи.

Для оцінки надійності ПС використовуються такі статистичні показники, як ймовірність і час безвідмовної роботи, можливість відмови й частота (інтенсивність) відмов. Оскільки як причини відмов розглядаються тільки помилки в програмі, які не можуть самоусуватися, то ПС варто відносити до класу  систем, що не оновлюють.

При кожному вияві нової помилки, як правило, здійснюється її локалізація й виправлення. При цьому накопичена до цього статистика про відмови втрачає своє значення, оскільки після внесення змін програма, власне кажучи, є новою програмою на відміну від тієї, яка до цього випробовувалася.

У зв'язку з виправленням помилок у ПС надійність, тобто її окремі атрибути, будуть увесь час змінюватися, як правило, у бік покращання. Отже, їхня оцінка буде мати тимчасовий і наближений характер. Тому виникає необхідність у використанні нових властивостей, адекватних реальному процесу вимірювання надійності, таких, як залежність інтенсивності виявлених помилок від числа прогонів програми й залежність відмов від часу функціонування ПС тощо.

Для багатьох систем (реального часу, радарні системи, системи безпеки, медичне устаткування з вбудованими програмами та ін.) надійність – головна цільова функція їх реалізації. Для таких  типів систем  висувають високі вимоги до надійності, такі, як відсутність помилок, вірогідність, безпека та ін.

Від 70-х років минулого століття було розроблено значну кількість моделей надійності ПЗ. Як основні критерії класифікації таких моделей вибрано два доволі прості предмети дослідження:

1) дослідження кількості відмов за певний період часу (вимір часу “в режимі настінного годинника” або вимір часу відносно виконання процесів пристроями комп’ютера);

2) дослідження часових проміжків між помилками.

Вважали, що моделі, класифіковані цим способом, взаємно не перетинаються і можуть містити часткові випадки в кожному дослідженні. Однією з перших складних моделей надійності ПЗ вважають модель Муси і Окумото. Для побудови цієї моделі використано набір атрибутів, зокрема:

• часовий проміжок;

• загальну кількість відмов, які можливо виявити за нескінченний або скінченний проміжок часу;

• розподіл кількості відмов, які відбулися за час *t* (розподіл Пуассона або біноміальний розподіл);

• клас відмови, або функціональна форма активності відмов за певний час (застосовується тільки для випробувань із скінченними часовими проміжками);

• тип відмови, або вигляд функції інтенсивності відмов протягом нескінченного проміжку часу (застосовується тільки для випробувань із нескінченними часовими проміжками).

З розвитком технологій розроблення і практичного застосування ПЗ схема класифікації моделей надійності ПЗ значно розширилася. Не тільки розроблено нові критерії класифікації, але й ускладнено саму структуру класифікації внаслідок об’єднання і перетину декількох критеріїв у різних моделях.

До найбільш вживаних методик і факторів надійності ПЗ належать: стадія життєвого циклу розроблення ПЗ (класифікація моделей залежить від етапу, на якому розраховується надійність ПЗ), можливість раннього прогнозування помилок, орієнтованість на інформацію або архітектуру (класифікацію роблять на підставі перевірки правильності вхідних / вихідних даних, або перевірки функціонального наповнення ПЗ), зростання надійності ПЗ в процесі виявлення та виправлення помилок тощо. Класифікація за такими факторами є найповнішою та дає можливість представити не тільки самі моделі, але і взаємозв’язок між ними (рис. 1).

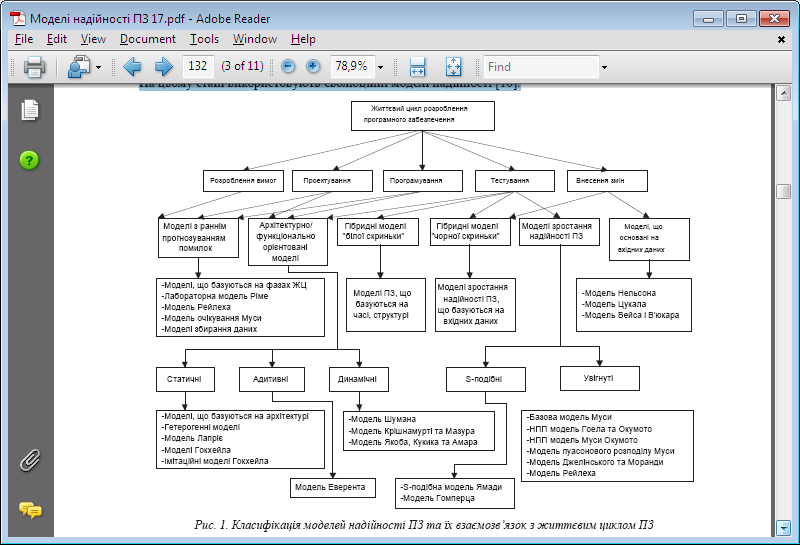
Кількість моделей надійності ПЗ сьогодні перевищує сотню і продовжує зростати.

Залежно від визначення поняття надійності ПЗ і глибини дослідження обирають різні критерії і характеристики моделей надійності. Тому наведена на рис. 1 схема класифікації моделей надійності ПЗ, як і розглянуті критерії, не є єдиною.

Схема, наведена на рис. 1, поділяє всі моделі надійності ПЗ на аналітичні та емпіричні.

Емпіричні моделі надійності, своєю чергою, поділяються на моделі складності і моделі, що визначають час, необхідний на “доведення” програми. Аналітичні моделі надійності ПЗ поділяють на динамічні (дискретні і неперервні) і статичні (за областю помилок, за областю даних).

Життєвий цикл ПЗ загалом зазвичай поділяють на такі етапи: специфікація вимог, проектування, кодування, тестування, експлуатація і супровід. Етап проектування може включати попереднє проектування і деталізоване проектування. Етап тестування може містити модульне, інтеграційне та регресійне тестування, тестування в умовах експлуатації. Етап супроводу може містити один або два цикли, кожен з яких має всі етапи стадії розробки. Згідно з ДСТУ -1999 (ІSO/IEC 12207:1995) основними процесами життєвого циклу ПЗ є процеси: замовлення, постачання, розроблення, експлуатації та супроводу. Своєю чергою, процес розроблення ПЗ містить такі дії: реалізацію процесу, аналіз системних вимог, проектування архітектури системи, аналіз вимог до ПЗ, проектування архітектури ПЗ, розроблення детального проекту ПЗ, кодування і тестування ПЗ, інтеграцію ПЗ, кваліфікаційне тестування ПЗ, системну інтеграцію, кваліфікаційне тестування системи, встановлення ПЗ та забезпечення приймання ПЗ.



*Рис. 1. Класифікація моделей надійності ПЗ та їх взаємозв’язок з життєвим циклом ПЗ*

На ранніх стадіях життєвого циклу ПЗ потрібна модель прогнозування надійності, оскільки даних про відмови немає. Моделі такого типу призначені для передбачення кількості помилок у програмі перед тестуванням, і в деяких літературних джерелах належать до детерміністичних (статичних) моделей надійності ПЗ. На етапі тестування показники надійності ПЗ покращуються завдяки відлагодженню програми. Модель зростання надійності на цьому етапі потрібна для оцінювання поточного рівня надійності, часу і ресурсів, потрібних для досягнення заданого рівня надійності ПЗ. Впродовж цього етапу оцінка надійності ґрунтується на аналізі цих відмов. Моделі такого типу належать до імовірнісних (динамічних) моделей надійності ПЗ.

Після введення програми в експлуатацію при визначенні її надійності необхідно враховувати додавання нових модулів, усунення старих модулів, усунення виявлених помилок, поєднання нового коду з попередньо написаним кодом, зміну середовища користувача, зміну апаратного забезпечення тощо. На цьому етапі використовують еволюційні моделі надійності.

За моделями, в основу яких покладено підрахунок відмов (динамічні моделі), припускають, що концептуально в програмі наявна скінченна кількість помилок. Враховуючи, що кількість помилок є цілим числом, за динамічними моделями обчислюють кількість початкових помилок на етапі відлагодження програми і кількість помилок, що залишились під час чи в кінці етапу відлагодження. Моделі підрахунку відмов використовують інтенсивність відмов як основну характеристику появи відмови. Залежно від типу моделі припускають, що інтенсивність відмов кожної помилки є або сталою функцією часу відлагодження, або випадковою змінною із заданим законом розподілу. Як тільки інтенсивність відмов, пов’язана з помилками певного типу, визначено, інтенсивність відмов програми загалом обчислюють як добуток кількості помилок, що залишились в програмі, на інтенсивність відмов, породжених помилкою кожного типу.

Під час етапу відлагодження кількість помилок, що залишились, змінюється. Одним із способів моделювання такого процесу відмов є представлення кількості помилок, що залишились, як стохастичного процесу.

Оцінка надійності ПЗ є все важливішою під час розроблення та тестування нових програмних продуктів. Перед тим, як нове ПЗ випустити в користування, його ретельно перевіряють на наявність помилок, які могли з’явитися в процесі розроблення ПЗ. Незважаючи на те, що виявлені помилки негайно видаляються, нові помилки можуть бути введені під час процесу відлагодження. Програмне забезпечення, яке містить помилки і випускається на ринок, несе високі ризики, пов’язані з відмовами. Відлагодження і тестування, з іншого боку, зменшує кількість помилок, але збільшує витрати на розроблення. Отже, існує необхідність визначити оптимальний час для зупинки тестування ПЗ. Під час тестування системи надійність є важливим критерієм при ухваленні рішення, коли саме потрібно випустити ПЗ. Декілька інших критеріїв, такі як кількість помилок, що залишилися, частота відмов, вимоги до надійності або загальна вартість системи можуть бути використані для визначення оптимального часу тестування.

**Формальне визначення моделей надійності ПС.**

З теоретичної точки зору під *надійністю*ПС розуміють здатність системи зберігати свої властивості (безвідмовність, відновлюваність) на певному рівні протягом фіксованого проміжку часу за певних умов експлуатації. Інакше кажучи, є імовірність того, що функціонуюча ПС у заданому середовищі не дасть відмов у інтервалі часу *(t,. t+k)* за умови, що остання відмова ПС й усунення відповідної помилки відбулися до моменту часу *t.* Чим більше часу працює система без відмов, тим вона надійніше.

Дослідження надійності проводиться за допомогою методів теорії ймовірностей, математичної статистики, відновлення систем й ін. Формально надійність можна визначити як таку ймовірність:

*P (i) = P*, якщо немає відмов у*i* прогонах програми на тестах,

*P(t) = P,* якщо немає відмов у інтервалі часу  *(0, t )* виконання програми.

Якщо при відмові працездатність системи підлягає відновленню в процесі експлуатації, то така система називається відновлюваною. Головними показниками такої системи є час безвідмовної роботи ПС і час її відновлення після відмови.

***Контрольні запитання для самоперевірки***.

1. Визначте поняття якість ПС і рівні моделі якості ПС.

2. Визначте характеристики якості ПС і їхнє призначення.

3. Які методи визначають показники якості?

4. Визначте  типи моделей надійності і їхній базис.

**Рекомендована література**

1. Бабенко Л.П., Лавріщева К.М*.* Основи програмної інженерії.– Навч. посібник.–К.: Знання, 2001.– 269с.
2. И. Соммервиль. Инженерия программного обеспечения,  
    6 изд. – И.д. "Вильямс", 2002.
3. Лавріщева К.М. Програмна інженерія. Електронний підручник: http://csc.knu.ua/uk/library/books/lavrishcheva-6.pdf
4. Лавріщева К.М. Програмна інженерія. – Підручник.–К.:Академперіодика, 2008.–415с.
5. Лаврищева Е.М. Сборочное программирование. Основы индустрии программных продуктов: монография / Е.М. Лаврищева, В.Н. Грищенко. — 2-е изд., дополн. и перераб. — К.: Наук. думка, 2009. — 372 с.
6. ДСТУ 2844–1994. Программные средства ЭВМ. Обеспечение качества. Термины и определения.
7. ДСТУ 2850–1994. Программные средства ЭВМ. Обеспечение качества. Показатели и методы оценки качества программного обеспечения.
8. ДСТУ 3230–1995. Управление качеством и обеспечение качества. Термины и определения.
9. Лаврищева Е.М.Методы программирования. Теория, инженерия, практика.–Киев: Наук. думка, 2006.–451с.

***Для самостійного вивчення***: Поглибити матеріал лекції за наданою літературою. Вивчення лекційного матеріалу та додаткових джерел. Розгляд запитань і виконання завдань для самостійної роботи, запропонованих на лекції.