| Must have рівень:   1. Зроби порівняння статичних та динамічних технік тестування. Наведи переваги та можливі обмеження при використанні кожної з них.  |  | Статистична техніка тестування | Динамічна техніка тестування | | --- | --- | --- | | Основна інформація | Перевірка коду до його безпосереднього запуску | Аналіз поведінки програми після запущення коду | | Перевага №1 | Мінімізація помилок в коді, повна перевірка коду | Можливість виправлення вже працюючого коду, з розумінням того, як саме він працює | | Перевага №2 | Підвищення якості розробки коду, з покращенням дизайну та підтримкою коду | Додаток тестується з точки зору користувача, що дає можливість отримати якісне ПЗ | | Перевага №3 (і т.д.) | Зниження загальної вартості на підтримку ПЗ, через раннє виявлення дефектів | Виявлення складних помилок, які могли бути пропущеними на етапі код рев’ю | | Обмеження №1 | Може займати багато часу, так як, в основному, виконується вручну | Складний процес, який займає багато часу | | Обмеження №2 | Перешкоджає виявленню вразливостей, представлених в середовищі виконання | Виконується після завершення кодування, знайдені баги виявляються пізніше в життєвому циклі розробки | | Обмеження №3 (і т.д.) | Є ризик дуже заглибитися в код, та нехтувати привабливістю користувацького інтерфейсу | Висока вартість проведення | | Висновок | Краща для перевірки коду, та для мінімізації помилок в ньому, та витрат на тестування | Краща, з точки зору характеристик створеного ПЗ (більш надійна, зручніша для користувача, продуктивніша), за рахунок перевірок вже працюючого функціоналу та системи | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Середній рівень:   1. Виконай завдання попереднього рівня. 2. Наступне твердження стосується покриття рішень:   *Коли код має одну ‘IF” умову, не має циклів (LOOP) або перемикачів (CASE), будь-який тест, який ми виконаємо, дасть результат 50% покриття рішень (decision coverage).*  Яке твердження є коректним?   * 1. Коректно. Будь-який тест кейс надає 100% покриття тверджень, таким чином покриває 50% рішень.   2. Коректно. Результат будь-якого тесту умови IF буде або правдими, або ні.   3. Некоректно. Один тест може гарантувати 25% перевірки рішень в цьому випадку.   4. Некоректно, бо занадто загальне твердження. Ми не можемо знати, чи є воно коректним, бо це залежить від тестованого ПЗ.   **P.s. Моя логіка працює так, що якщо ми маємо тільки одну умову, без додаткових, то у нас всього 2 варіанти - або воно спрацює, або ні. Тобто, сходячи з того, що 100% - наш максимум, і в нас 2 варіанти результатів тесту - вірогідність кожного з них - 50%.**   1. Є псевдокод: Switch PC on -> Start MS Word -> IF MS Word starts THEN -> Write a poem -> Close MS Word.   Скільки тест кейсів знадобиться, щоб перевірити його функціонал?   * 1. 1 – для покриття операторів, 2 – для покриття рішень   2. 1 – для покриття операторів, 1 – для покриття рішень   3. 2 – для покриття операторів, 2 – для покриття рішень   4. 2 – для покриття операторів, 1 – для покриття рішень   **P.s. Моя логіка в цьому працює так, що 2 покриття операторів - це що MS Word або запуститься, або ні. А покриття рішень - якщо він запуститься, то ми маємо написати розповідь, а потім закрити його.**  **Але я можу помилятися, бо ця тема ще не дуже добре засвоїлася у моїй голові. То ж, як завжди, буду рада коментарям.:)**   1. Скільки потрібно тестів для перевірки тверджень коду:     1. 2    2. 1    3. 3    4. 4       **P.s. Або я дуже ускладнюю собі життя, і тут просто треба було написати:**  **1) If P+Q більше 100 - Print “Large”, і нам потрібно було б підставити такі значення P та Q, які будуть відповідати умові, та подивитися на реакцію системи, тому що вона повинна реагувати так, як задано в наших умовах, тобто, написати “Large”**  **2) Підставити хибні значення, результат котрих не співпадає з нашою умовою, перевірити реакцію системи, вона не повинна писати “Large”, якщо умова не виконується**  **3) If P більше 50 - Print “P Large”, маємо підставити таке значення P, яке відповідає умові (не менше 51), та подивитися на реакцію системи, тому що вона повинна реагувати так, як задано в наших умовах, тобто, написати “P Large”**  **4) Підставити хибне значення P, де результат не співпадає з нашою умовою (від 50 і нижче), перевірити реакцію системи, вона не повинна писати “P Large”, якщо умова не виконується** |
| Програма максимум:   1. Виконай завдання двох попередніх рівнів. 2. Продовжуємо розвивати стартап для застосунку, який дозволяє обмінюватися фотографіями котиків.   Є алгоритм:  *Запитай, якого улюбленця має користувач.*  *Якщо користувач відповість, що має кота, то запитай, яка порода його улюбленця: «короткошерста чи довгошерста?»*  *Якщо клієнт відповість «довгошерста», то запитай: «ви бажаєте отримати контакти найближчого грумера?»*  *Якщо клієнт відповість «так», то скажи: «Надайте адресу найближчої котячої перукарні»*  *Інакше*  *Скажи: «Запропонуй магазин з товарами по догляду за шерстю»*  *Закінчити*  *Інакше*  *Скажи «Запропонуй обрати магазин із зоотоварами»*  *Закінчити*  *Якщо клієнт не має кота*  *Скажи “Коли вирішите завести улюбленця – приходьте”*  *Закінчити*  Завдання:   1. Намалюй схему алгоритму (в інструменті на вибір, наприклад, у вбудованому Google Docs редакторі, [figjam](https://www.figma.com/figjam/) чи [miro](https://miro.com/)) 2. Який потрібен мінімальний набір тест-кейсів, щоб переконатися, що всі запитання були поставлені, всі комбінації були пройдені та всі відповіді були отримані? |

**Намалюй схему алгоритму (в інструменті на вибір, наприклад, у вбудованому Google Docs редакторі,** [**figjam**](https://www.figma.com/figjam/) **чи** [**miro**](https://miro.com/)**)**

<https://www.figma.com/file/HNIHithmmI9zgDdHsgLLkM/Sankova.-Lesson-9?node-id=0%3A1&t=QP5S3Nzb8YhHa5bZ-1> (якщо будуть проблеми з доступом - напиши мені, будь ласка:)

**2.** **Pre - conditions:** Зайшли до нашого додатку, як зареєстрований користувач, заходимо на сторінку опитування.

1. Перевірити реакцію системи, коли користувач відповідає, який у нього домашній улюбленець.

**Expected results:**

Якщо користувач обирає “кота” - система має видати йому “Короткошерста чи довгошерста порода?”

Якщо користувач відповість “не кота” - система має видати йому “Коли вирішите завести улюбленця - приходьте” та закінчити опитування.

1. Перевірити реакцію системи, коли користувач відповідає “Довгошерста”.

**Expected results:**

Система має запропонувати обрати магазин із зоотоварами та закінчити опитування.

Система має запропонувати магазин з товарами по догляду за шерстю та завершити опитування.

Система має спитати, чи бажає користувач отримати контакти найближчого грумера?

1. Перевірити реакцію системи, коли користувач відповідає на питання “чи бажаєте отримати контакти найближчого грумера?” “так”

**Expected results:**

Система має надати адресу найближчої котячої перукарні.