|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 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 | Є методикою тестування програмного забезпечення, при якій ПЗ тестується без запуску коду. Воно являє собою процес або техніку, які виконуються для пошуку потенційних дефектів в програмному забезпеченні. Це також процес виявлення і усунення помилок і дефектів в різних супровідних документах, таких як специфікації вимог до програмного забезпечення і т. ін. | виконується шляхом фактичного використання програми і визначення того, чи працює функціональність так, як очікується | | Перевага №1 | Менша вартість виправлення багів | Забезпечує функціональність продукту | | Перевага №2 | Забезпечує більш ширше охоплення,аніж динамічне, за коротший проміжок часу. | Включає в себе як функціональне так і нефункціональне тестування | | Перевага №3 (і т.д.) | Мета – запобігання дефектів програмного забезпечення | Мета – пошук і усунення багів | | Обмеження №1 | Процес статичного тестування може займати багато часу | Велика вартість виправлення багів | | Обмеження №2 | Перешкоджає виявленню вразливостей, представлених в середовищі виконання. | Виявляється менше дефектів,аніж при статичному тестуванні | | Обмеження №3 (і т.д.) | у в основному він виконується вручну. | Виконується після деплоя коду | | Висновок | незважаючи на те, що статичне тестування вимагає багато часу на бурхливі дискусії та зустрічі, все ж варто витратити час на запобігання появи дефектів на останніх етапах розробки продукту. Тому статичне тестування по праву вважається важливим кроком на шляху до розробки ПЗ без помилок. | важливість динамічного тестування також величезна. Завдяки безпосередньому виконанню тестів програмного забезпечення (перевірки функціональної поведінки, продуктивності, надійності та інших важливих аспектів) команда може перевірити і підтвердити якість і ефективність ПЗ. | |
| 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. (Переходом вважається будь який умовний перехід чи цикл (IF – THEN – ELSE, WHILE-DO, FOR). Покриття вираховується як відношення кількості переходів, покритих тестами до загальної кількості переходів в коді програми. І, до речі, 100% Decision покриття гарантує 100% Statement покриття, але не навпаки) 3. Коректно. Результат будь-якого тесту умови IF буде або правдими, або ні. 4. Некоректно. Один тест може гарантувати 25% перевірки рішень в цьому випадку. 5. Некоректно, бо занадто загальне твердження. Ми не можемо знати, чи є воно коректним, бо це залежить від тестованого ПЗ.      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      1. Скільки потрібно тестів для перевірки тверджень коду:     1. 2    2. 1    3. 3    4. 4 |
| 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? |

