

# Manual QA

Техніки тест-дизайну. Частина 1

# Manual QA

## План уроку

- Black box
  - Класи еквівалентності (Equivalence Class)
  - Граничні значення (Boundary Value)
  - Таблиці Рішень (Decision Tables)
- White box
  - Тестування операторів (Statement testing)
  - Тестування умов (Condition testing)
  - Тестування рішень / гілок (Decision/branch testing)
- Як писати гарні тест-кейси
- Практика написання простих документів

Що було минулого разу

# Manual QA

## План уроку

- Процес тестування. SDLS та STLS
- Документи планування тестування
  - Тест-план
  - Стратегія тестування
- Документи проведення тестування
  - Практичний Тест-план
  - Чек-лист
  - Тест-кейс
  - Тест-сценарій
  - Тестова матриця
- Документи звітності про тестування
  - Тест-звіт
  - Звіт про дефект

## Техніки тест-дизайну

# Manual QA

## Test Design

**Тест-дизайн** – це один з етапів процесу розробки програмного забезпечення, етап вигадування, розробки та впорядкування тестів у набори, на основі аналізу вимог та функціональності продукту.

**Техніка тест-дизайну** – це метод проектування тестів.

# Manual QA

## Test Design. Думаємо разом



У ящику може бути

1. Нічого
2. М'яч
3. Кіт

Як перевірити? (Тести)

1. Тести
2. Тести
3. Тести
4. Тести

# Manual QA

## Test Design

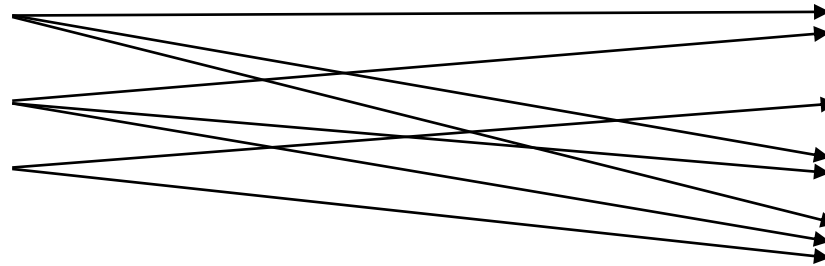


У ящику може бути

1. Нічого
2. М'яч
3. Кіт

Як перевірити? (Тести)

1. Підняти та зважити ящик
2. Послухати
3. Потрясти
4. Відкрити





# Manual QA

## Test Design. Цілі

### 1. Забезпечити глибоке покриття

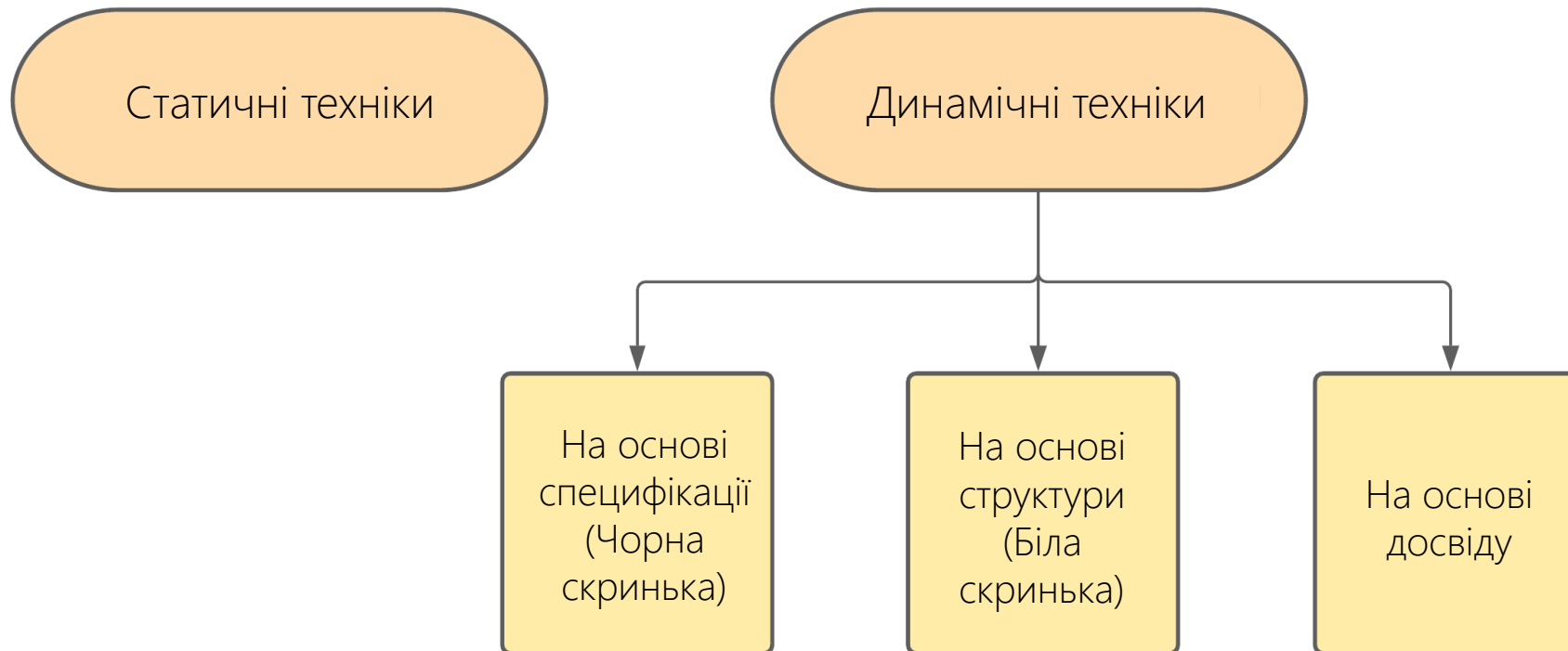
Вигадати тести, які виявлять найбільш серйозні помилки продукту. Так, ми можемо вигадувати тести, які знаходять несерйозні помилки, але тоді тестування буде неефективним.

### 2. Мінімізувати кількість тестів

що необхідні для знаходження більшості серйозних помилок. Ми можемо вигадати стільки тестів, скільки не в змозі будемо виконати. Тому перед розробниками тестів завжди стоїть завдання зберегти ефективність тестів (тобто, їх здатність виявляти серйозні помилки) без збільшення їх числа.

# Manual QA

## Test Design. Техніки



## Test Design. Статичні техніки

**Статичні методи розробки тестів** — це методи тестування, які включають тестування без виконання коду або програмного застосунку.

В основному статичне тестування пов'язане з різного роду переглядами та аналізом коду, тестів та іншої проектної документації.

### Ручні методи статичного дизайну (Manual Static Design Techniques)

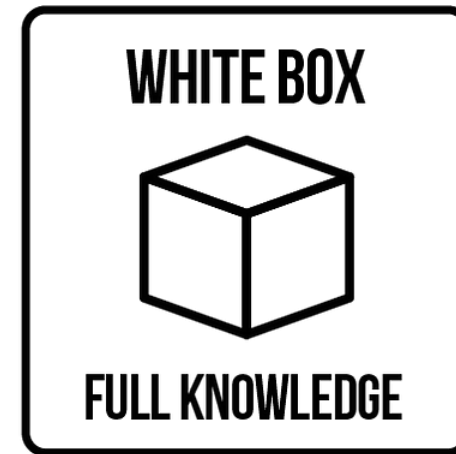
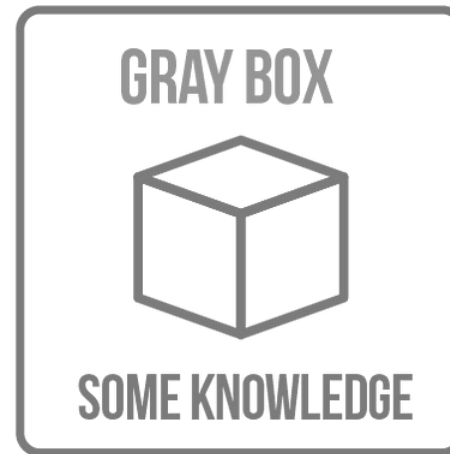
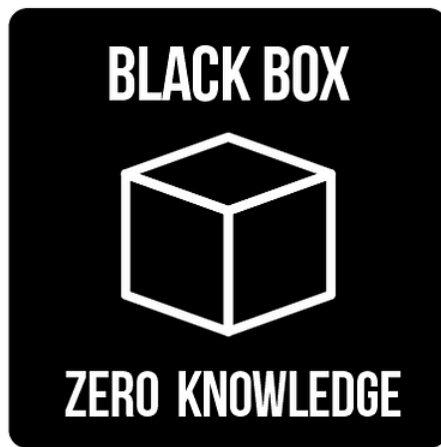
- перегляд (Walkthrough)
- неформальна рецензія (Informal reviews)
- технічна рецензія (Technical reviews)
- аудит (Audit)
- інспекція (Inspection)
- менеджмент-рецензія (Management reviews)

### Інструментальні методи статичного дизайну (Static Design Techniques)

- статичний аналіз коду (Static analysis of code)
- відповідність коду стандартам (Compliance with coding standards)
- аналіз метрик коду (Analysis of code metrics)

# Manual QA

## Test Design. Динамічні техніки



## Test Design. На основі специфікації

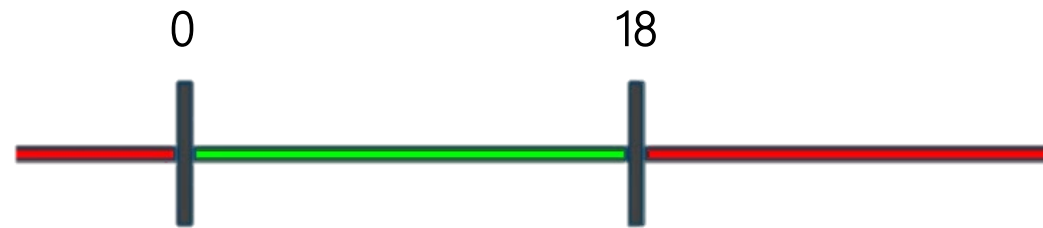


- Класи еквівалентності (Equivalence partitioning)
- Граничні значення (Boundary value analysis)
- Таблиця прийняття рішень (Decision tables)
- Діаграма переходів станів (State transition testing)
- Попарне тестування (Pairwise Testing)
- Користувацькі сценарії (Use case testing)

## Класи еквівалентності (Equivalence partitioning)



Це розділення тестових даних на групи (класи), які застосунок обробляє однаково.



Важливо протестувати хоч одне значення кожного класу.

Важливо перевіряти як «прийнятні», так і «неприйнятні» класи еквівалентності.

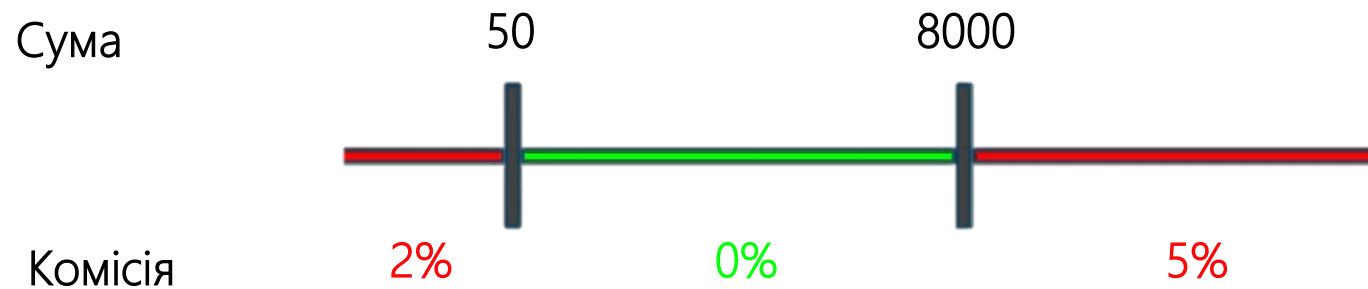
## Класи еквівалентності. Приклад



### Завдання:

Визначити класи еквівалентності для зняття готівки в банкоматі:

Зняття до 50 грн йде з комісією в 2%, а більше 8000 з комісією в 5% .



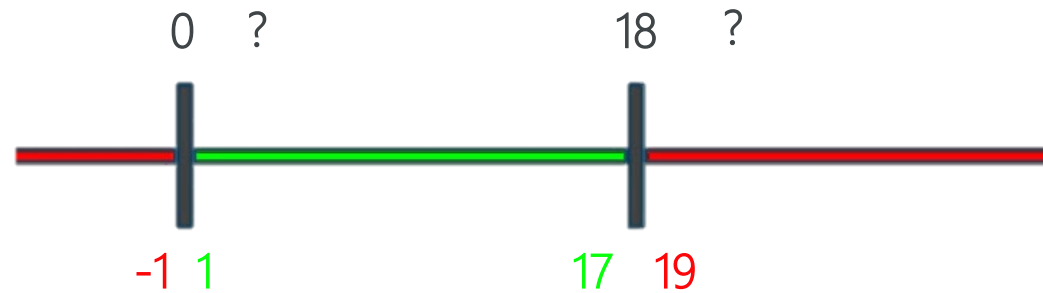
Важливо протестувати хоча б одне значення кожного класу.

Важливо перевіряти як «прийнятні», так і «неприйнятні» класи еквівалентності.

## Граничні значення (Boundary value analysis)



Як бути із значеннями на межах класів?



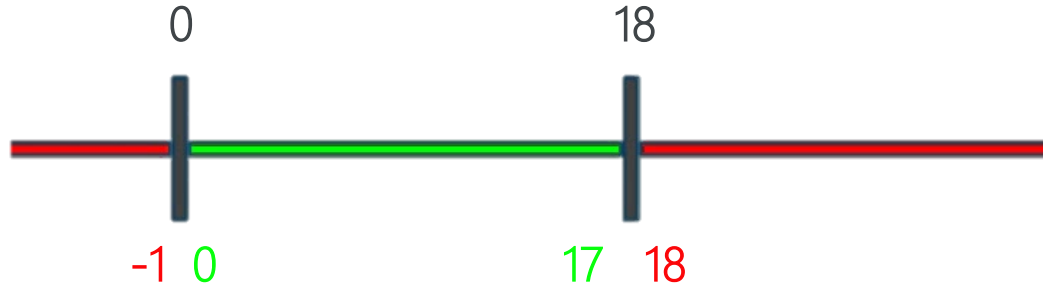


## Тестування Граничними значеннями. Приклад

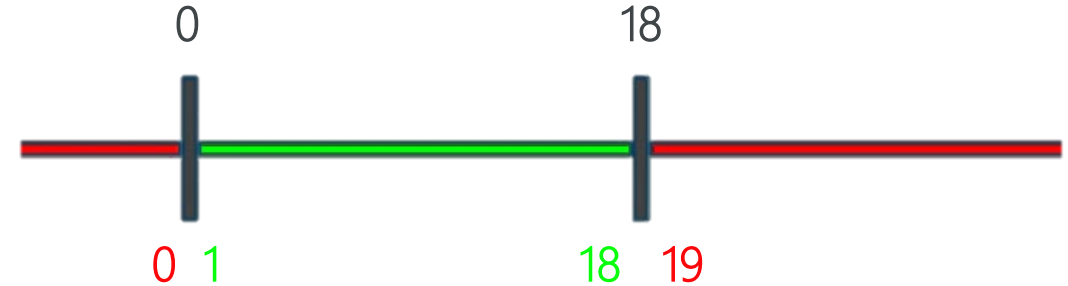


Це перевірка значеннями, що знаходяться на границях класів еквівалентності.

Як ви думаєте які вони? Та до яких класів належать?



Знижка в 15% застосовується лише до неповнолітніх



Знижка в 15% застосовується лише при покупці до 19 одиниць товару

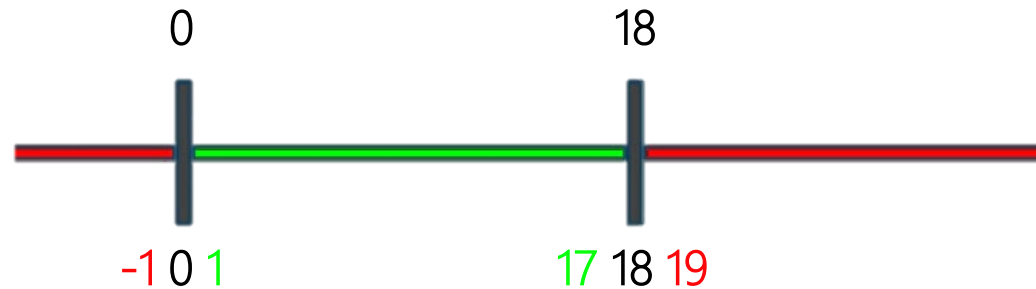


Значення границь завжди включені в один із класів еквівалентності.



## Граничні значення (Boundary value analysis)

- Граничні значення – це мінімальне та максимальне значення класу еквівалентності.
- Вони мають велику ймовірність помилки.
- Перевірка граничних умов вважається методами розширених класів еквівалентності.

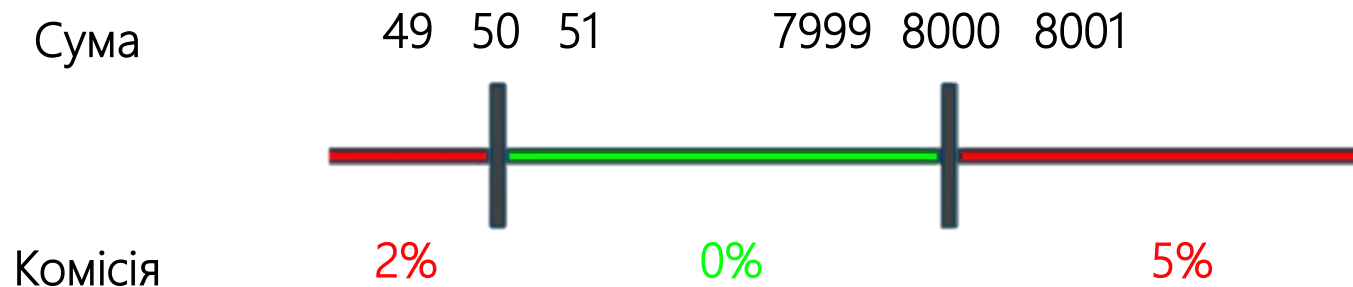




## Аналіз граничних значень (Boundary value analysis)

### Завдання:

Визначити граничні значення класів еквівалентності для зняття готівки в банкоматі:  
Зняття до 50 грн йде з комісією в 2%, а більше 8000 з комісією в 5%.



Важливо протестувати хоч одне значення кожного класу.

Важливо перевіряти як «прийнятні», так і «неприйнятні» класи еквівалентності.

## Таблиця прийняття рішень (Decision tables)



Таблиця прийняття рішень за стандартом ISTQB – метод проектування тестів шляхом «чорної скриньки», в якому тестові приклади призначені для виконання комбінацій вхідних даних та причин, показаних у таблиці рішень.

Простими словами, така таблиця наочно показує взаємозалежності умов та операцій у вимогах і при спільному виконанні, що призводять до конкретного результату.

## Таблиця прийняття рішень (Decision tables)



		Правила							
Умови	Принтер друкує	Ні	Ні	Ні	Ні	Так	Так	Так	Так
	Червоне світло блимає	Так	Так	Ні	Ні	Так	Так	Ні	Ні
	Принтер розпізнається комп'ютером	Ні	Так	Ні	Так	Ні	Так	Ні	Так
Дії	Перевірте кабель живлення			✓					-
	Перевірте кабель принтер-комп'ютер	✓		✓					-
	Переконайтеся, що програмне забезпечення принтера встановлено	✓		✓		✓		✓	-
	Перевірити/замінити чорнила	✓	✓				✓		-
	Перевірте наявність застрягання паперу		✓		✓				-



## Таблиця прийняття рішень (Decision tables). Практика

**Завдання:** скласти таблицю прийняття рішень для самокерованого автомобіля для проходження світлофору.

Move	Stop	Prepear	Give control to a driver
------	------	---------	--------------------------

Можливі стани автопілоту



Можливі комбінації кольорів.





## Таблиця прийняття рішень (Decision tables). Практика

Приклад таблиці для розв'язання задачі зі світлофором:

#	Green	Yellow	Red	Move	Stop	Prepear	Give control to a driver
1	×	×	×				Yes
2	×	×	✓		Yes		
3	×	✓	×			Yes	
4	×	✓	✓		Yes	Yes	
5	✓	×	×	Yes			
6	✓	×	✓				Yes
7	✓	✓	×				Yes
8	✓	✓	✓				Yes

## Test Design. На основі структури



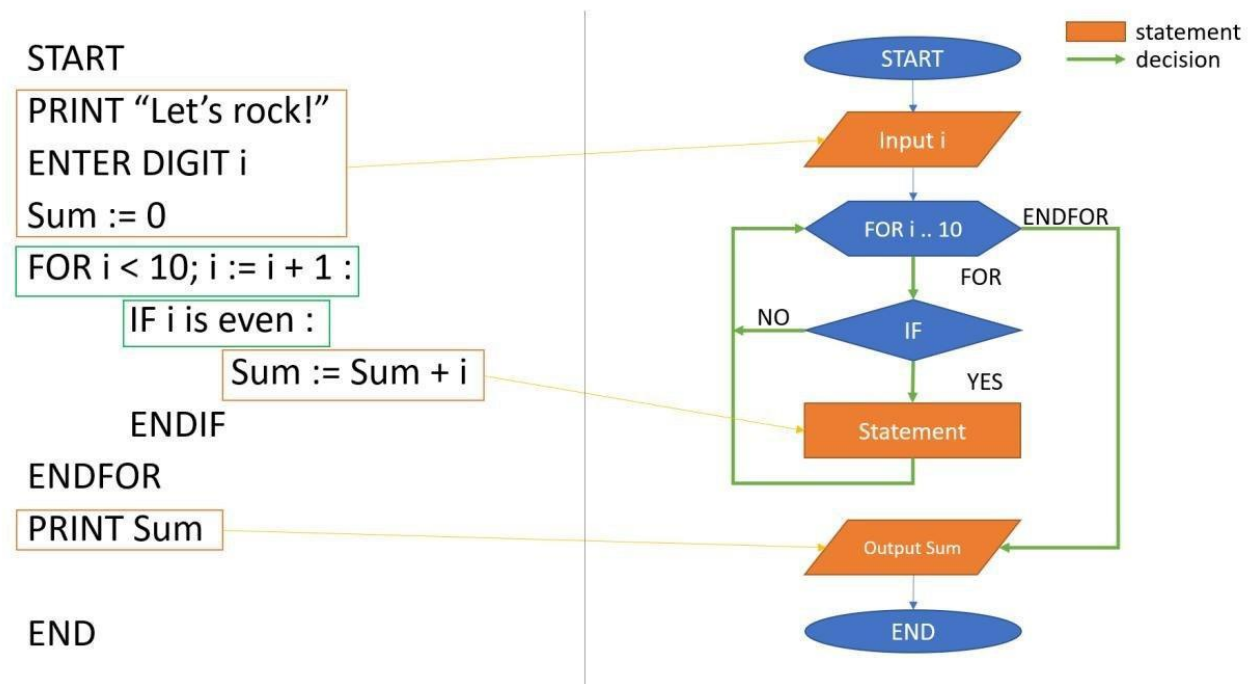
- Тестування операторів (Statement testing)
- Тестування умов (Condition testing)
- Тестування рішень/гілок (Decision/branch testing)





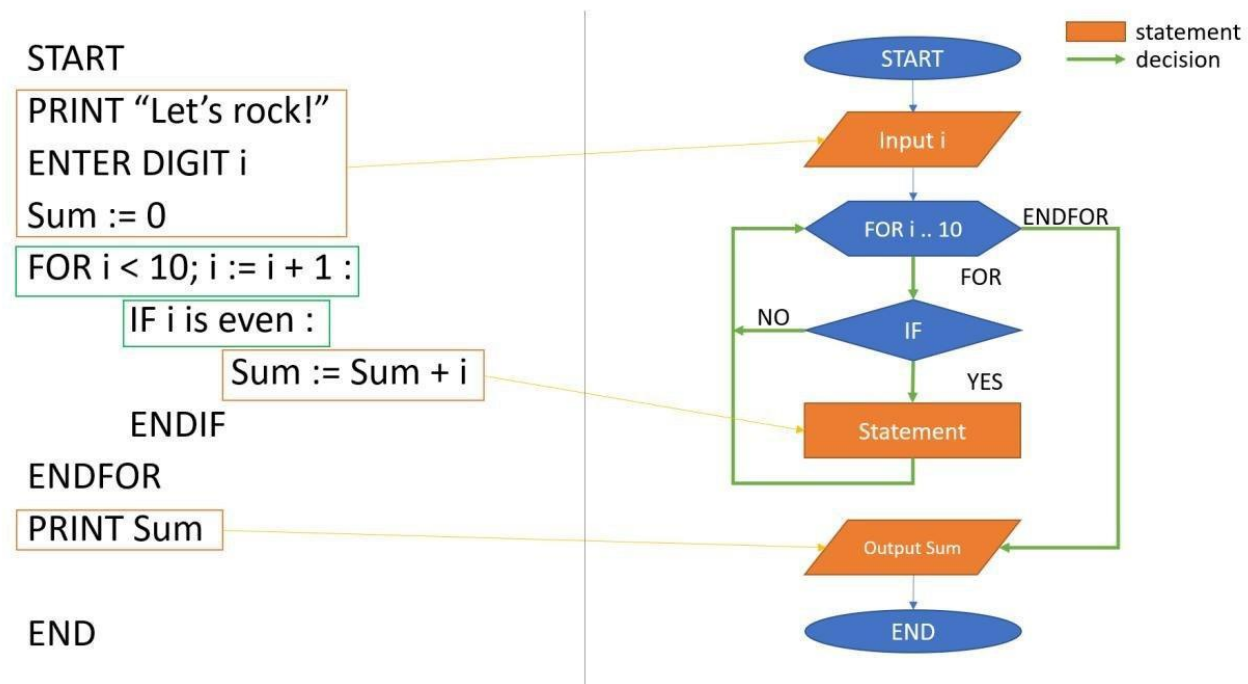
## Тестування операторів (Statement testing)

Statement Testing (Тестування операторів) – це коли тестові скрипти розробляються для виконання виразів коду, а покриття – це міра підрахунку рядків коду, пройдених тестовим скриптом.



## Тестування умов (Condition testing)

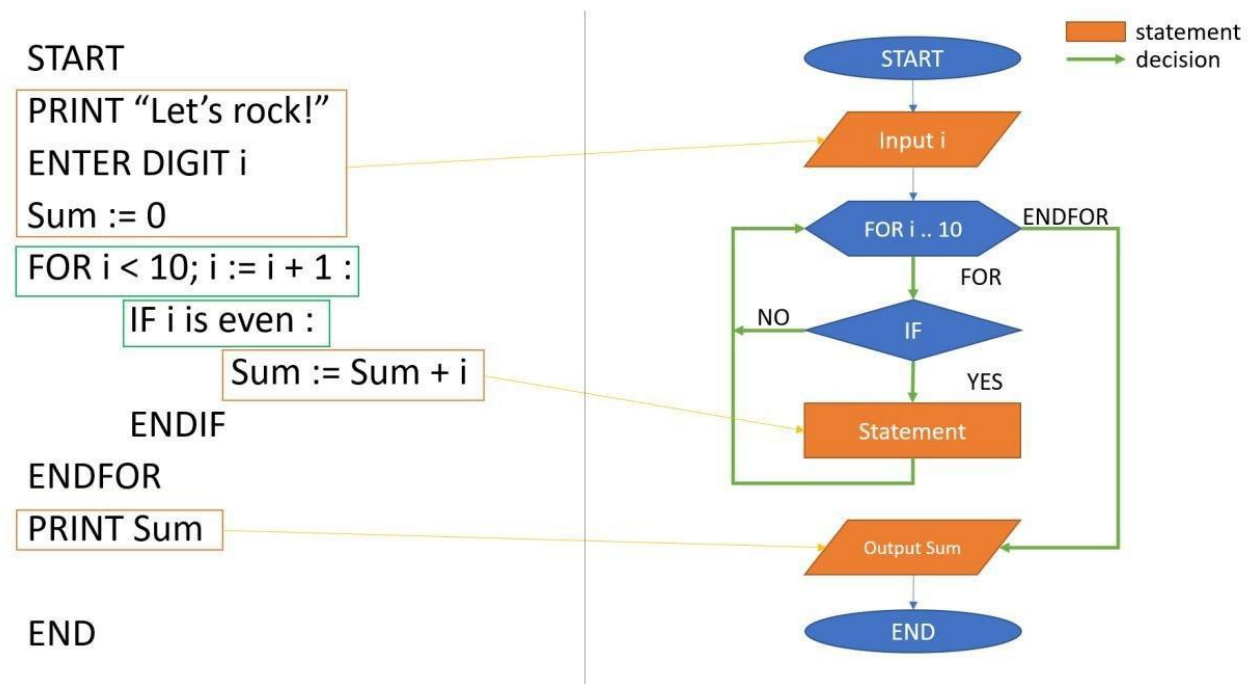
**Condition Testing (Тестування умов)** – включає перевірку окремих умов для обох результатів – TRUE та FALSE. Таким чином, для отримання 100-відсоткового охоплення умов необхідно виконати кожну умову як для ІСТИНИХ, так і для ХИБНИХ результатів.





## Тестування рішень/гілок (Decision/branch testing)

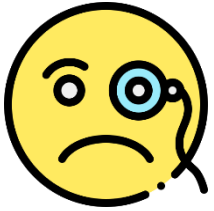
Decision testing/branch testing (Тестування рішень/гілок) – це вже більш високий рівень, який охоплює всі типи ієрархій (if-else, switch операторів, виклики методів). Вимірює відсоток точок рішень, виконуються в розрахунку від загальної кількості умов у аплікації.



## Властивості якісних тестів

# Manual QA

## Баланс між простотою та складністю



### Запуск застосунку

1. Запустити застосунок.

1. Застосунок запускається.

### Повторна конвертація

Приготування:

\* Створити в корені будь-якого диска три окремі папки для вхідних файлів, вихідних файлів, файлу журналу.

\* Підготувати набір з кількох файлів максимально підтримуваного розміру підтримуваних форматів з підтримуваними код ріками, а також декількох файлів допустимого розміру, але неприпустимого формату.

1. Запустити застосунок, вказавши в параметрах відповідні шляхи з приготування до тесту (ім'я файлу журналу - довільне).

2. Скопіювати в папку для вхідних файлів кілька файлів допустимого формату.

3. Перемістити конвертовані файли з папки з результатуючими файлами у папку для вхідних файлів.

4. Перемістити сконвертовані файли з папки з результатуючими файлами у папку з набором файлів для тесту.

5. Перемістити всі файли з папки з набором файлів для тесту в папку для вхідних файлів.

6. Перемістити сконвертовані файли з папки з результатуючими файлами до папки для вхідних файлів.

2. Файли поступово переміщуються з вхідної у вихідну папку, в консолі та файлі журналу з'являються повідомлення про успішну конвертацію файлів.

3. Файли поступово переміщуються з вхідної у вихідну папку, в консолі та файлі журналу з'являються повідомлення про успішну конвертацію файлів.

5. Файли поступово переміщуються з вхідної у вихідну папку, в консолі та файлі журналу з'являються повідомлення про успішну конвертацію файлів ДОПУСТИМОГО формату та повідомлення про ігнорування файлів неприпустимого формату.

6. Файли поступово переміщуються з вхідної у вихідну папку, в консолі та файлі журналу з'являються повідомлення щодо успішної конвертації файлів ДОПУСТИМОГО формату та повідомлення про та Ігнорування файлів неприпустимого формату.

### Переваги простих тест-кейсів:

- легке виконання;
- зрозумілі для новачків;
- спрощення діагностики помилки;
- роблять наявність помилки очевидною.

### Переваги складних тест-кейсів:

- вище шанси знайти баг;
- користувачі, як правило, використовують складні сценарії;
- програмісти самі рідко перевіряють такі варіанти.

# Manual QA

## Баланс між специфічністю та загальністю



1. У полі А ввести 10. 2. У полі В ввести 15. 3. Натиснути кнопку «Скласти». 4. Перевірити значення у полі С.	4. Значення у полі С дорівнює 25.
--	-----------------------------------

1. Перевірити, що програма підсумовує два числа коректно.	1. Підсумовує коректно.
---	-------------------------

### У разі надмірної специфічності:

- при повторних виконаннях тест-кейсу завжди виконуватимуться строго одні й ті самі дії, що знижує можливість виявити помилку;
- зростає час створення та підтримки тест-кейсу.

### У разі надмірної загальності:

- тест-кейс складний для тестувальників-початківців;
- тест-кейс цілком може залишитися невиконаним з низки об'єктивних та суб'єктивних причин.

## Баланс між специфічністю та загальністю



### Додавання А і В

1. У полі А ввести коректне ціле число.
2. У полі В ввести коректне ціле число.
3. Натиснути кнопку "Скласти".
4. Перевірити значення поля С.
5. Повторити кроки 1-4 для значень: 0, максимального та мінімального допустимих значень.

4. Значення поля С дорівнює сумі А та В.

Тут ми не прив'язані до **конкретних** значень:

- знаємо, як перевірити результат;
- скорочуємо час написання та підтримки тест-кейсу посиланням на кроки 1-4;
- перерахували значення, які становлять особливий інтерес для нас.

## Рекомендації написання тестів



- **Описуйте складні кроки** або області фокусу тестового прикладу:
  - якщо ви тестуєте пошук – будьте детальні;
  - якщо ви використовуєте пошук, щоб отримати продукт, який ви збираєтеся тестувати, напишіть 1-м рядком;
- **пропускайте деталі простих** або часто використовуваних кроків;
- вкажіть **приклади/умови/готові запити** тестових даних, якщо для їх пошуку потрібні додаткові знання/дії;
- **пам'ятайте** про всіх, **хто** користуватиметься **вашими тестами** – користувачі/команда автоматизації/нові члени команди - для них може мати сенс додати додаткові деталі.



# Manual QA

Практика

# Manual QA

## Практика

1. Напишіть чек-лист. Інтернет-магазин робить важливу промоакцію – протягом одного робочого дня з 8 ранку і до 22 вечора на сайті мають бути знижки на різні категорії товарів у різний час:
  - До 12 на взуття, з 12 і до 18 на курточки, а з 18 до 22 – на штани.
  - Знижка на взуття 15%, на штани - 10%, а на куртки - 30%.
2. Створіть таблицю рішень для тестування системи продажів у кінотеатрі, згідно з нею на фільми діє знижка – 30% у таких випадках:
  - Ранкова сесія в будь-який день – 15%
  - Усі сеанси у глядацький день – 30%
  - Пільгові категорії відвідувачів (студенти, пенсіонери, інваліди) за умови надання довідок – 30%
  - Знижки підсумовуються, за виключенням знижки у глядацький день

# Manual QA

Що ми сьогодні вивчили

# Manual QA

## План уроку

- Black box
  - Класи еквівалентності (Equivalence Class)
  - Граничні значення (Boundary Value)
  - Таблиці Рішень (Decision Tables)
- White box
  - Тестування операторів (Statement testing)
  - Тестування умов (Condition testing)
  - Тестування рішень / гілок (Decision/branch testing)
- Як писати гарні тест-кейси
- Практика написання простих документів

# Manual QA

## Підсумки

Що одне, найголовніше, ви дізнались сьогодні?

# Manual QA

Дякую за увагу! До нових зустрічей!

# Інформаційний відеосервіс для розробників програмного забезпечення

