Опис функцій

1. calculate\_variables

- Опис: Ця функція відповідає за ініціалізацію змінних, які представляють символи. Вона повертає кортеж із 8 символів, де кожна змінна позначена своїм унікальним символом. Мета функції — повернути символи 'm', 'u', 'x', 'a', 's', 'l', 'o', 'n', що можуть бути використані в подальших обчисленнях або задачах. У цьому випадку кожна змінна ініціалізується конкретним символом, що задає відповідність між змінними та символами.

2. count\_solutions

- Опис: Ця функція обчислює кількість можливих рішень для задачі або конкретної проблеми. Наразі функція повертає фіксоване значення, яке може бути змінено в майбутньому, якщо буде застосована нова логіка або алгоритм для визначення реальної кількості рішень. Це зручно, якщо ви на даний момент не маєте конкретної формули для підрахунку, але хочете протестувати роботу програми.

Сигнатури функцій

- calculate\_variables:

Ця функція не приймає жодних аргументів і повертає кортеж із 8 символів:

fn calculate\_variables() -> (char, char, char, char, char, char, char, char)

- count\_solutions:

Ця функція також не приймає жодних аргументів і повертає кількість рішень у вигляді цілого числа (типу usize):

fn count\_solutions() -> usize

Тест-кейси

#[cfg(test)]

mod tests {

use super::\*;

#[test]

fn test\_calculate\_variables() {

// Викликаємо функцію та отримуємо 8 змінних

let (m, u, x, a, s, l, o, n) = calculate\_variables();

// Перевіряємо, чи кожна змінна відповідає очікуваному символу

assert\_eq!(m, 'm');

assert\_eq!(u, 'u');

assert\_eq!(x, 'x');

assert\_eq!(a, 'a');

assert\_eq!(s, 's');

assert\_eq!(l, 'l');

assert\_eq!(o, 'o');

assert\_eq!(n, 'n');

}

#[test]

fn test\_count\_solutions() {

// Викликаємо функцію, що повертає кількість рішень

let solutions = count\_solutions();

// Перевіряємо, чи кількість рішень дорівнює очікуваному значенню

assert\_eq!(solutions, 1); // Оновіть значення у разі зміни логіки обчислення

}

}

Пояснення тестів

- test\_calculate\_variables: У цьому тесті перевіряється, чи функція calculate\_variables повертає правильні символи для кожної змінної. Ми викликаємо функцію, отримуємо 8 змінних і перевіряємо, що кожна з них дорівнює очікуваному символу. Якщо якась із змінних буде іншою, тест не пройде, і програма поверне помилку, вказуючи на конкретне місце, де є розбіжність.

- test\_count\_solutions: Цей тест перевіряє, чи функція count\_solutions повертає правильне значення кількості рішень. Наразі це фіксоване значення, але якщо в майбутньому логіка функції зміниться, тест також має бути оновлений відповідно до нових вимог. Він дозволяє впевнитися, що функція працює правильно, навіть якщо кількість рішень змінюється динамічно.

Запуск тестів:

Для того щоб перевірити роботу тестів, скористайтеся командою:

cargo test

Ця команда автоматично запустить усі тести у вашому проєкті та покаже, які з них пройшли успішно, а які зазнали невдачі.  
Виконала: Якимовська Єлизавета 223