Мутационный критерий и пример, иллюстрирующий технику работы

Мутационный критерий — это подход в тестировании программного обеспечения, который позволяет оценить качество набора тестов путём внесения в код искусственных мелких изменений (мутаций) и проверки их обнаружения. Цель — определить, достаточно ли надёжен набор тестов для выявления ошибок.

Основные понятия:

- Мутации — мелкие ошибки в программе (например, замена арифметических операторов, изменение значений переменных, перестановка индексов в массивах и т. п.).
- Мутанты — версии программы, отличающиеся друг от друга внесёнными мутациями.

Метод мутационного тестирования заключается в следующем:

1. В разрабатываемую программу P вносят мутации, создавая программы-мутанты P1, P2 и т. д.
2. Программа P и её мутанты тестируются на одном и том же наборе тестов (X,Y).
3. Если на наборе (X,Y) подтверждается правильность программы P и при этом выявляются все внесённые в программы-мутанты ошибки, то набор тестов (X,Y) соответствует мутационному критерию, а тестируемая программа объявляется правильной.
4. Если некоторые мутанты не выявили всех мутаций, необходимо расширять набор тестов и продолжать тестирование.

Пример применения мутационного критерия:

Рассмотрим программу, которая вычисляет неотрицательную степень n числа x.

Исходная программа P:

```
static public double PowerNonNeg(double x, int n) {
   double z=1;
   if (n>0) {
      for (int i=1; n-1>=i; i++) {
         z = z*x;
      }
   }
   else {
      Console.WriteLine("Ошибка! Степень числа n должна быть больше 0.");
   }
   return z;
}
```

Создаются две программы-мутанта:

- P1: изменено начальное значение переменной z с 1 на 2.
- P2: изменено начальное значение переменной i с 1 на 0 и граничное значение индекса цикла с n на n-1.

Набор тестов: (X,Y) = {(x=2, n=3, y=8), (x=999, n=1, y=999), (x=0, n=100, y=0)}.

Результаты:

- При запуске тестов выявляются все ошибки в программах-мутантах.

- Также обнаруживается ошибка в основной программе, где в условии цикла вместо n стоит n-1.

Вывод: набор тестов (X,Y) соответствует мутационному критерию, а тестируемая программа считается правильной. Если бы некоторые мутанты не были обнаружены, потребовалось бы расширить набор тестов.

Дополнительные аспекты:

- Для создания мутантов используются мутационные операторы — правила преобразования исходного кода, которые имитируют типичные ошибки программистов.
- Если мутант не обнаруживается тестами, это может указывать на слабость тестового покрытия или на то, что мутация не влияет на функциональность (например, если изменённый код никогда не выполняется).
- Метрика mutation score отражает соотношение убитых и сгенерированных мутантов. Чем она выше, тем надёжнее тесты.