МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Програмна інженерія та інформаційні технології управління»

Звіт з лабораторної роботи №2

З предмету «Якість і тестування програмного забезпечення»

Варіант №19

Виконав:

Студент групи КН-36а

Рубан Ю. Д.

Перевірив:

Копп А. М.

Харків – 2018

Тема: Тестирование методами “белого ящика”

Цель работы: Знакомство с методами тестирования логики программы (методами покрытия), создание формализованного описания результатов тестирования и проведение тестирования на реальном примере программного кода.

Ход работы:

Тестирование методом белого ящика (также: прозрачного, открытого, стеклянного ящика; основанное на коде или структурное тестирование) – метод тестирования программного обеспечения, который предполагает, что внутренняя структура/устройство/реализация системы известны тестировщику. Мы выбираем входные значения, основываясь на знании кода, который будет их обрабатывать. Точно так же мы знаем, каким должен быть результат этой обработки. Знание всех особенностей тестируемой программы и ее реализации – обязательны для этой техники. Тестирование белого ящика – углубление во внутренне устройство системы, за пределы ее внешних интерфейсов.

Преимущества:

– тестирование может производиться на ранних этапах: нет необходимости ждать создания пользовательского интерфейса;

– можно провести более тщательное тестирование, с покрытием большого количества путей выполнения программы.

Недостатки:

– для выполнения тестирования белого ящика необходимо большое количество специальных знаний

– при использовании автоматизации тестирования на этом уровне, поддержка тестовых скриптов может оказаться достаточно накладной, если программа часто изменяется.

Перед выбором метода тестирования и написанием теста обозначим на диаграмме активности все ветви алгоритма.

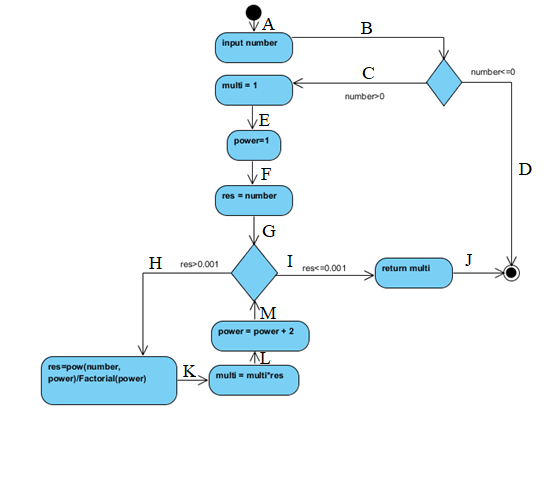


Рисунок 1 – Диаграмма активности с обозначениями ветвей алгоритма

Исходя из особенностей алгоритма для тестирования был выбран метод комбинаторного покрытия, так как данный метод позволяет провести достаточное количество тестов для выявления ошибок. Этот метод требует создания такого числа тестов, чтобы все возможные комбинации результатов условия в каждом решении выполнялись по крайней мере один раз. Набор тестов, удовлетворяющих критерию комбинаторного покрытия условий, удовлетворяет также и критериям покрытия решений, покрытия условий и покрытия решений/условий.

По этому критерию в алгоритме, который создан в предыдущей лабораторной работе, должны быть покрыты тестами следующие 4 комбинаций:

1. number = 0
2. number 0
3. number < 0
4. number > 0

Эти комбинации можно проверить двумя тестами:

- number 0{покрывает а, c};

- number > 0 {покрывает c, d}

Сведем результаты тестов в таблицу:

|  |  |  |  |
| --- | --- | --- | --- |
| Тест | Ожидаемый результат | Фактический результат | Результат тестирования |
| number = 0 | 0 | Ошибка вводимых значений | успешно |
| number = 4 | result = 0.00079 | result = 0.00079 | неуспешно |

Вывод:

После проведения теста разработанного алгоритма были найдена одна ошибка. К ошибке привела тестовая ситуация с данными, не соответствующими требуемым критериям.