

2 Лабораторная работа №2

Исследование способов структурного тестирования программного обеспечения

2.1. Цель работы

Исследовать основные подходы к структурному тестированию программного обеспечения. Приобрести практические навыки построения графа потоков управления и определения независимых ветвей программы.

2.2. Постановка задачи

Был выдан вариант 14.

Варианты заданий соответствуют заданиям по лабораторной работе №1. По варианту задаются требования к программам. Для каждой из них необходимо: 1) написать программу, выполняющую заданные действия; 2) построить граф потоков управления; 3) вычислить цикломатическое число для построенного графа потоков управления; 4) определить независимые ветви программы.

2.3. Ход выполнения работы

2.3.1. Граф потоков управления

По составленной в прошлой лабораторной работе программе были составлены графы потоков управления (рисунки 2.1 и 2.2).

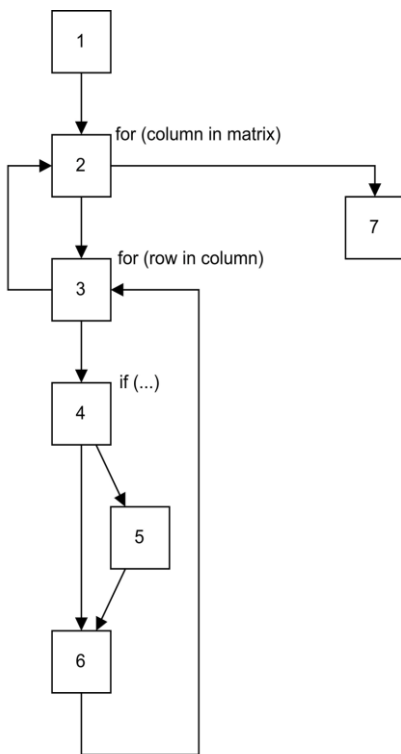


Рисунок 2.1 — Граф потоков управления для программы в задании 1

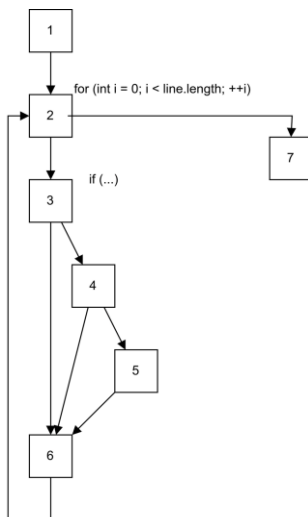


Рисунок 2.2 — Граф потоков управления для программы в задании 3

Для задачи 2 невозможно составить граф по причине того, что метод, выдающий конечный результат, состоит из вызова одной стандартной функции.

2.3.2. Результаты расчета цикломатического числа

Для первой задачи цикломатическое число рассчитано в соответствии с формулой (2.1).

$$C(G_1) = 9 - 7 + 2 = 4 \quad (2.1)$$

Для третьей задачи — по формуле (2.2).

$$C(G_3) = 9 - 7 + 2 = 4 \quad (2.2)$$

2.3.3. Независимые ветви управления

Для первой задачи были получены следующие независимые ветви:

1. 1, 2, 7;
2. 1, 2, 3, 2, 7;
3. 1, 2, 3, 4, 5, 6, 3, 2, 7;
4. 1, 2, 3, 4, 6, 3, 2, 7.

Для третьей задачи получены следующие независимые ветви:

1. 1, 2, 7;
2. 1, 2, 3, 6, 2, 7;
3. 1, 2, 3, 4, 6, 2, 7;
4. 1, 2, 3, 4, 5, 6, 2, 7.

Выводы по результатам работы

При выполнении данной лабораторной работы были исследованы способы тестирования программного обеспечения, а именно тестирование ветвей. Были получены практические навыки нахождения цикломатического числа, позволяющего вычислить количество независимых ветвей, построения графа потоков управления, позволяющего найти эти независимые ветви.