## Практическое занятие №6. Оценка покрытия программы и проекта

Тестирование программы P по некоторому критерию C означает покрытие множества компонентов программы P M = {m1...mk} по элементам или по связям

T = {t1...tn} — кортеж неизбыточных тестов ti.

Тест ti неизбыточен, если существует покрытый им компонент mi из M(P,C), не покрытый ни одним из предыдущих тестов t1...ti-1. Каждому ti соответствует неизбыточный путь pi — последовательность вершин от входа до выхода.

V(P,C) — сложность тестирования P по критерию C — измеряется max числом неизбыточных тестов, покрывающих все элементы множества M(P,C)

DV(P,C,T) — остаточная сложность тестирования P по критерию C — измеряется max числом неизбыточных тестов, покрывающих элементы множества M(P,C), оставшиеся непокрытыми, после прогона набора тестов T. Величина DV строго и монотонно убывает от V до 0.

TV(P,C,T) = (V-DV)/V — оценка степени тестированности P по критерию C.

Критерий окончания тестирования TV(P,C,T) L, где (0 L 1). L — уровень оттестированности, заданный в требованиях к программному продукту.

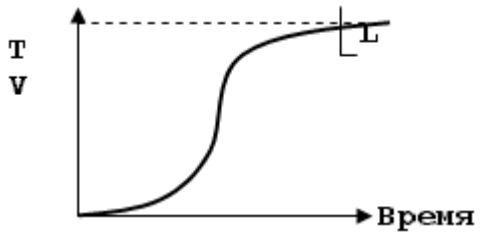


*Рис. 4.1. Метрика оттестированности приложения*

Рассмотрим две модели программного обеспечения, используемые при оценке оттестированности.

Для оценки степени оттестированности часто используется УГП — управляющий граф программы. УГП многокомпонентного объекта G (Рис. 4.2, Пример 4.4), содержит внутри себя два компонента G1 и G2, УГП которых раскрыты.

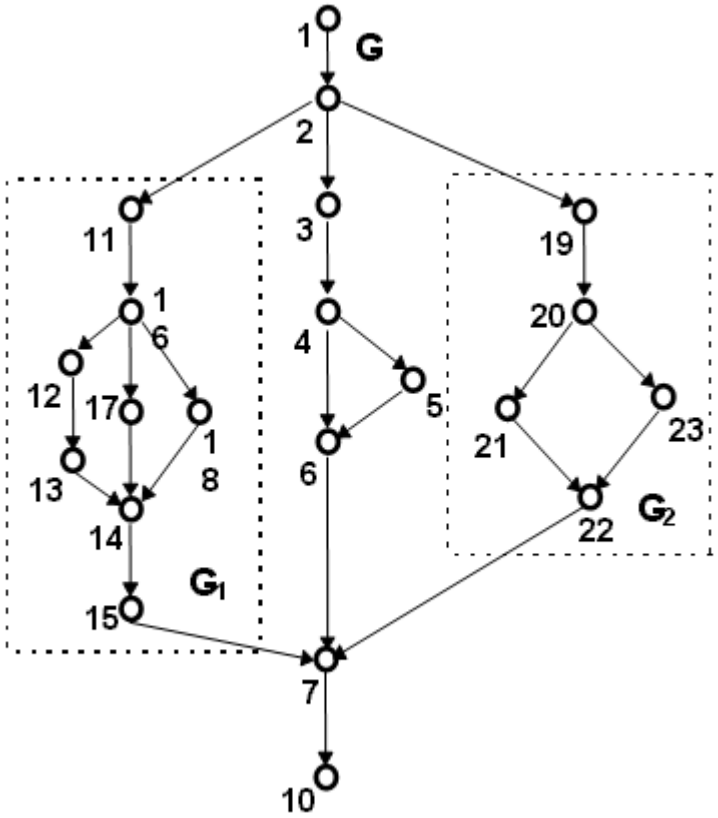


*Рис. 4.2. Плоская модель УГП компонента G*

В результате УГП компонента G имеет такой вид, как если бы компоненты G1 и G2 в его структуре специально не выделялись, а УГП компонентов G1 и G2 были вставлены в УГП G. Для тестирования компонента G в соответствии с критерием путей потребуется прогнать тестовый набор, покрывающий следующий набор трасс графа G (Пример 4.1):

P1(G) = 1-2-3-4-5-6-7-10;

P2(G) = 1-2-3-4-6-7-10;

P3(G) = 1-2-11-16-18-14-15-7-10;

P4(G) = 1-2-11-16-17-14-15-7-10;

P5(G) = 1-2-11-16-12-13-14-15-7-10;

P6(G) = 1-2-19-20-23-22-7-10;

P7(G) = 1-2-19-20-21-22-7-10;

Пример 4.1. Набор трасс, необходимых для покрытия плоской модели УГП компонента G

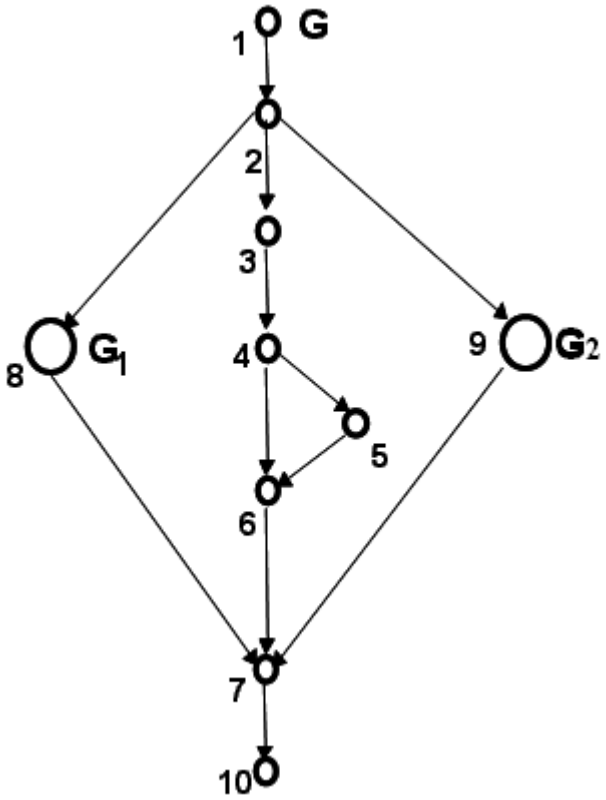


*Рис. 4.3. Иерархическая модель УГП компонента G*

УГП компонента G, представленный в виде иерархической модели, приведен на Рис. 4.3, Пример 4.5. В иерархическом УГП G входящие в его состав компоненты представлены ссылками на свои УГП G1 и G2 (Рис. 4.4, Пример 4.5)

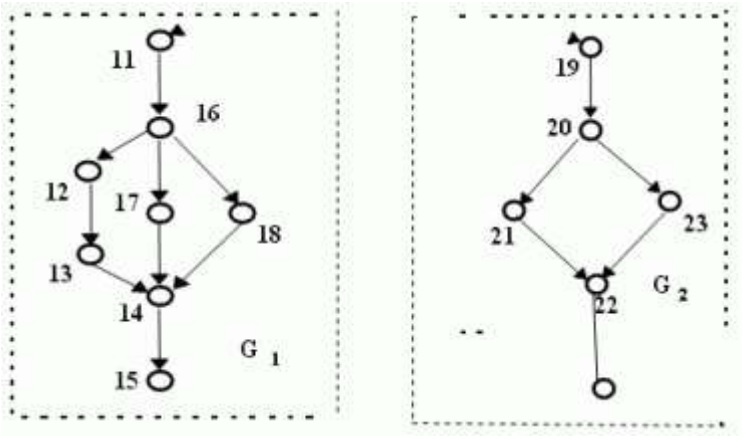


*Рис. 4.4. Иерархическая модель: УГП компонент G1 и G2*

Для исчерпывающего тестирования иерархической модели компонента G в соответствии с критерием путей требуется прогнать следующий набор трасс (Пример 4.2):

P1(G) = 1-2-3-4-5-6-7-10;

P2(G) = 1-2-3-4-6-7-10;

P3(G) = 1-2-8-7-10;

P4(G) = 1-2-9-7-10.

Пример 4.2. Набор трасс, необходимых для покрытия иерархической модели УГП компонента G

Приведенный набор трасс достаточен при условии, что компоненты G1 и G2 в свою очередь исчерпывающе протестированы. Чтобы обеспечить выполнение этого условия в соответствии с критерием путей, надо прогнать все трассы Пример 4.3.

P11(G1)=11-16-12-13-14-15;

P21(G2)=19-20-21-22;

P12(G1)=11-16-17-14-15;

P22(G2)=11-16-18-14-15;

P13(G1)=19-20-23-22.

Пример 4.3. Набор трасс иерархической модели УГП, необходимых для покрытия УГП компонентов G1 и

## Методика выполнения лабораторной работы

1. Проанализировать программу, реализующую заданный алгоритм обработки данных – по индивидуальным вариантам из сборника задач. Номер задачи соответствует номеру студента в журнале учёта посещаемости +23.
2. Отобразить алгоритм решения задачи в виде схемы программы ( см. /2,3/).
3. Обозначить буквами или цифрами ветви алгоритма
4. Отладить программу и провести тестирование программного продукта рассмотренными методами.
5. Выписать пути алгоритма, которые должны быть проверены тестами для рассматриваемого метода тестирования.
6. Записать тесты, которые позволят пройти по путям алгоритма,.
7. Протестировать разработанную Вами программу. Результаты оформить в виде таблиц (таблицы 2.1-2.4). Оценить степень оттестированности программы Вашими тестами по критерию TV(P,C,T)
8. Оформить отчет по лабораторной работе.

## 4 Содержание отчета

1. Цель работы.
2. Программа решения поставленной Вам задачи.
3. Схема программы (см. пп.2,3).
4. Таблицы тестирования программы (п.7) и оценка степени оттестированности.

6. Выводы по результатам тестирования (целью тестирования является обнаружение ошибок в программе).