

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

**Український державний університет науки і технологій**

Кафедра «Комп’ютерні інформаційні технології»

**Лабораторна робота №4**

**з дисципліни «Програмні засоби загального користування»**

**на тему: «Оформлення науковоі статті. Набір математичних формул.»**

Виконав:

студент гр. ПЗ2011 Кулик С. В.

Прийняла:

Самойлов С. П.

Дніпро, 2022

**УДК 68 1.3**

**И.Г. ГУБАРЬ, С.И. ГУБАРЬ**

**О ЗАДАЧЕ ПРОЕКТИРОВАНИЯ ОПТИМАЛЬНЫХ ТЕСТОВ ДЛЯ СТРУКТУРИРОВАНЫХ ПРОГРАММ**

**Предложена модель планирования тестов для структурированой программы с учётом ограничений на сложность тестирования.**

Пусть задана структуррированая програма реализующая N функций f1, f2, …, fN и содержащая n структур ветвления. Будем считать, что для каждой функции известнно: распределение вероятностей её использования – α4; штраф за единичное появление сбоя этой функции Q1; сложность реализации функции – R1.

Введм обозначения: K1 - количество тестов для f1, P1 - вероятность появления ошибки при реализации функции. Тогда математическое ожидание штрафа при однократной реализации программы можно записать в виде

(1)

Построим управляющий граф этой программы. Узлами графа являются узлы начала и конца (по одному), n узлов ветвления и n узлов соединения. Ветвями графа являются линейные блоки программы. Переименуем произвольным образом ветви Vj, I ≤ j ≤ N1, узлы ветвления U1, U’1 и вычислительные пути f0, I ≤ s ≤ N.

Будем считать, что каждый вычислительный путь реализует свою функцию; что для каждой дуги известны: её сложность ƞ и вероятность проявления ошибки в этой ветви приреализации пути q1 до начала тестирования, что сложность построения и реализации теста любого блока пропорциональна сложности єтого блока.

обозначим через bj  - вероятность проявления ошибки в ветви Vj предиката узла U1, через ξ – матрицу взаимодействия ветвей и путей, т.е. ξj=1 при Vj ∈ f1 и ξj=0 при Vj ∉ f1.

Задачу оптимального планирования тестов можно сформулировать слкдующим способом:

(2)

где *R* – максимально возможная суммарная сложность тестов. В такой постановке неизвестными являются величины K1, через которые мы должны выразить все остальные параметры. Можно выделить две задачи тестирования: прямую – по заданным K1 определить и обратную – по информации о определить K1. Соединив выход программы с её ветвью V0, получим сильносвязанній граф.

Обозначим через nj(k) количество активизаций j ветви при общем количестве реализаций программы, равном K. Записав для каждого узла U1 уравнение Кирхгофа, получим систему из 2n+2 уравнений с 3n+3 неизвестными. Следовательно, система содержит n свободных неизвестных (способ их выделения основан на построении основного дерева графа).

Таким образом если задать K и c1, то определяются все nj(k). Положив последовательно с1 = 1, мы тем самым выделим конкретный вычислительный путь.

(3)

Будем считать, что для всех j1bj,0 известны – это эффективность обработки ветви до тестирования. Разоьъем множество допустимых значений ветви (Dj) на непересекающиеся подмножества даных, каждое из которых может вызвать появление ошибки в ветви и каждое из подмножеств равновероятное по использованию. В этих предроложениях мы приходим к простейшой линейной форме учета результатов тестирования

(3)

*,*

где bj зависит отмодуля и свойств даных.

Значит,

Следовательно из (3) можно записать

(4)

где ns – суммарное количество активизаций ветви.

Учитывая (4), окнчательно получим

где сомножителями используются только ветви, принадлижащие f1. Если Pi и bj,ns малы, то приближенно можно записать так:

(5)

Если функции f1 слабо связаны, то ,

где зависит от fi.

**Библиографические ссылки**

1. **Кнут, Д. Э.** Исскуство програмирования для ЭВМ [Текст] / **Д.Э. Кнут** В 3т. – М.:Мир.1976. – Т.1. – 730 с.

*Получено 07.07.1997*

**Висновок**

Під час виконання лабораторної роботи друкував наукову статтю, ознайомився з правилами оформлення статті, навчився користуватися редактором формул, закріпив навички форматування тексту.