

Міністерство освіти і науки України
Запорізький національний технічний університет

кафедра програмних засобів

РЕФЕРАТ

з дисципліни «Якість програмного забезпечення» на тему:

"Метрики похідні від цикломатичної складності"

Виконала:

студентка групи КНТ-415

В.С. Хохлова

Прийняла:

Г.В. Табунщик

2018

ЗМІСТ

1. МЕТРИКИ СКЛАДНОСТІ ПОТОКУ КЕРУВАННЯ ПРОГРАМИ.....	3
2. ІНТЕРВАЛЬНА МЕТРИКА ДЖ. МАЙЕРСА.....	6
3. ТОПОЛОГІЧНА МІРА ЧЕНА.....	7
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	8

1. МЕТРИКИ СКЛАДНОСТІ ПОТОКУ КЕРУВАННЯ ПРОГРАМИ

Метрики складності потоку керування програм – великий клас метрик, заснований на аналізі керуючого графа програми.

Розглянемо визначення керуючого графу програми і спосіб його побудови.

Нехай представлена певна програма. Для даної програми будується орієнтований граф, що містить лише один вхід і один вихід, при цьому вершини графа співвідносять з тими ділянками коду програми, в яких є лише послідовні обчислення і відсутні оператори розгалуження і циклу, а дуги співвідносять з переходами від блоку до блоку і гілками виконання програми. Умова при побудові даного графа: кожна вершина досяжна з початкової, і кінцева вершина досяжна з будь-якої іншої вершини.

Цикломатична складність програми - структурна (або топологічна) міра складності комп'ютерної програми. Міра була розроблена Томасом Дж. Маккейб в 1976 році.

При обчисленні цикломатичної складності використовується граф потоку керування програми. Найпоширенішою оцінкою, заснованою на аналізі отриманого графа, є цикломатичне число Мак-Кейба, що визначається як:

$$V(G) = e - n + 2p, \quad (1.1)$$

де e – кількість дуг, n – кількість вершин, p – кількість компонентів зв'язності.

Кількість компонентів зв'язності графа можна розглядати як кількість дуг, що необхідно додати для перетворення графа в сильно зв'язний. Сильно зв'язним називається граф, будь-які дві вершини якого взаємно досяжні. Для графів коректних програм, тобто графів, що не мають недосяжних від точки входу ділянок і "висячих" точок входу і виходу, сильно зв'язний граф, як

правило, виходить шляхом замикання дугою вершини, що означає кінець програми, на вершину, що позначає точку входу в цю програму. По суті $V(G)$ визначає кількість лінійно незалежних контурів в сильно доладному графі. Так що в коректно написаних програмах $p=1$, і тому формула (1.1) для розрахунку цикломатичної складності набуває вигляду:

$$V(G)=e-n+p, \quad (1.2)$$

Розглянемо приклад графу потоку керування програми (рис. 1.1). Програма починає виконуватися з червоного вузла, потім йдуть цикли (після червоного вузла йдуть дві групи по три вузла). Вихід з циклу здійснюється через умовний оператор (нижня група вузлів) і кінцевий вихід з програми в синьому вузлі. Для цього графа $E = 9$, $N = 8$ і $P = 1$, цикломатична складність програми дорівнює $9 - 8 + 2 \times 1 = 3$.

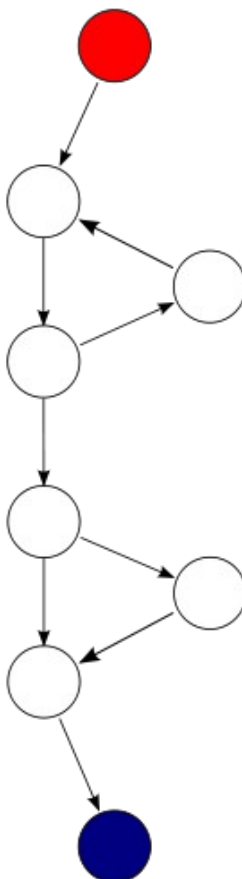


Рисунок 1.1 – Граф управління потоком простої програми

Як правило, при обчисленні цикломатичної складності логічні оператори не приймаються до уваги, допускається також спрощений підхід, згідно з яким власне побудова графа не проводиться, а показник визначається на підставі підрахунку кількості операторів керуючої логіки (*if*, *switch* і т. д.) і можливої кількості шляхів виконання програми. Метрика цикломатичної складності може бути розрахована для модуля, методу та інших структурних одиниць програми.

В процесі аналізу значень показника для окремих структурних елементів можна виявити елементи з високим значенням показника (наприклад, нормальне значення показника для методу – не вище 5-7), що свідчить про складність їх керуючої логіки і, відповідно, високих трудовитрат на розробку, тестування і супровід.

Обчислення метрики в ході реалізації проекту (а при детальному проектуванні воно можливо ще на цьому етапі, не чекаючи стадії кодування) дозволяє вчасно визначити найбільш складні, що супроводжуються високими ризиками, структурні одиниці та вжити заходів щодо усунення ризиків за рахунок внесення корективів.

Метрика цикломатичної складності стала основою для створення похідних і якісно нових метрик, таких як: інтервальна метрика Дж. Майерса, метрика У. Хансена, метрика Пивоварського, міра Чена, метрики Харрисона и Мейджела, метод граничних значень та інші.

2. ІНТЕРВАЛЬНА МЕТРИКА ДЖ. МАЙЕРСА

Крім своїх різновидів, метрика цикломатическая складності стала основою для створення похідних і якісно нових метрик, таких як інтервальна метрика Дж. Майерса (G. Mayers) - розраховується як різниця значень цикломатическая складності і числа окремих умов плюс одиниця.

В якості оцінки використовується інтервал:

$$[V(G), V(G) + h], \quad (2.1)$$

де h для простих предикатів дорівнює нулю, а для n -місних $h = n-1$ (n -місцевий предикат залежить від n змінних).

Даний метод дозволяє розрізняти різні за складністю предикати, однак на практиці він майже не застосовується. Що стосується аналізу бінарного коду, метрика Майерса не має помітних переваг перед простий цикломатическая складністю при вирішенні будь-якої з трьох завдань.

3. ТОПОЛОГІЧНА МІРА ЧЕНА

Складність програми визначається числом перетинів кордонів між областями, 9 утвореними графом програми.

Цей метод можна застосовувати тільки до структурованим програмами з послідовним з'єднанням керуючих конструкцій. Для неструктурованих програм міра Чена істотно залежить від умовних і безумовних переходів. У цьому випадку можна вказати верхню і нижню границі. Верхня – є $m+1$, де m – кількість логічних операторів при їх взаємній вкладеності. Нижня – дорівнює 2.

Якщо граф потоку керування має тільки одну компоненту зв'язності, міра Чена збігається з цикломатическая мірою МакКейба.

Стосовно до побудови профілю складності трас виходить, що жодна з розглянутих метрик не є для цього достатньою. Необхідні гібридні метрики, що поєднують заходи складності потоку керування з кількісними заходами. Як варіант, така метрика може бути побудована на основі метрики Пивоварського або метрики граничних значень, і відносної міри Джілбі або ж розміру базового блоку.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- 1) Інженерія якості програмного забезпечення: навчальний посібник / Г.В. Табунщик, Р.К. Кудерметов, Т.І. Брагіна. – Запоріжжя: ЗНТУ, 2013. – 180 с.;
- 2) Метрики сложности кода [Електронний ресурс]. – Режим доступу: http://www.ispras.ru/preprints/docs/rep_25_2013.pdf;
- 3) Программный код и его метрики [Електронний ресурс]. – Режим доступу: <https://habr.com/company/intel/blog/106082/>;
- 4) Automatic Measurement of Source Code Complexity [Електронний ресурс]. – Режим доступу: <http://www.diva-portal.org/smash/get/diva2:1019963/FULLTEXT02.pdf>;
- 5) Software code metrics [Електронний ресурс]. – Режим доступу: <https://www.viva64.com/en/a/0045/#ID0EAOAE>.