# Лабораторна робота № 6.

**Програмний код і його метрики**

**Короткі відомості**

Однією з тем в програмуванні, до яких інтерес періодично то з'являється, то зникає, є питання метрик коду програмного забезпечення. У великих програмних середовищах час від часу з'являються механізми підрахунку різних метрик. Хвилеподібний інтерес до теми так виглядає тому, що до цих пір в метриках не придумано головного - що з ними робити. Тобто навіть якщо якийсь інструмент дозволяє добре підрахувати деякі метрики, то що з цим робити далі найчастіше незрозуміло. Звичайно, метрики - це і контроль якості коду (не пишемо великі і складні функції), і «продуктивність» (в лапках) програмістів, і швидкість розвитку проекту.

**Кількісні метрики.** Перш за все, слід розглянути кількісні характеристики вихідного коду програм (на увазі їх простоти). Найелементарнішій метрикою є кількість рядків коду (SLOC). Дана метрика була спочатку розроблена для оцінки трудовитрат за проектом. Однак через те, що одна і та ж функціональність може бути розбита на кілька рядків або записана в один рядок, метрика стала практично непридатною з появою мов, в яких в один рядок може бути записано більше однієї команди. Тому розрізняють логічні і фізичні рядки коду. Логічні рядки коду - це кількість команд програми. Даний варіант опису так само має свої недоліки, так як сильно залежить від використовуваної мови програмування і стилю програмування.

Крім SLOC до кількісних характеристик відносять також:

* кількість порожніх рядків,
* кількість коментарів,
* відсоток коментарів (відношення числа рядків, що містять коментарі до загальної кількості рядків, виражене у відсотках),
* середнє число рядків для функцій (класів, файлів),
* середнє число рядків, що містять вихідний код для функцій (класів, файлів),
* середнє число рядків для модулів.

Іноді додатково розрізняють оцінку стилістики програми (F). Вона полягає в розбитті програми на n рівних фрагментів і обчисленні оцінки для кожного фрагмента за формулою Fi = SIGN (Nкомм.i / Ni - 0,1), де Nкомм.i - кількість коментарів в i-му фрагменті, Ni - загальна кількість рядків коду в i-му фрагменті. Тоді загальна оцінка для всієї програми буде визначатися таким чином: F = СУМА Fi. [2]

Також до групи метрик, заснованих на підрахунку деяких одиниць в коді програми, відносять метрики Холстеда. Дані метрики засновані на наступних показниках:

n1 - число унікальних операторів програми, включаючи символи-роздільники, імена процедур і знаки операцій (словник операторів),

n2 - число унікальних операндів програми (словник операндів),

N1 - загальне число операторів в програмі,

N2 - загальне число операндов в програмі,

n1 '- теоретичне число унікальних операторів,

n2 '- теоретичне число унікальних операндів.

З огляду на введені позначення, можна визначити:

n = n1 + n2 - словник програми,

N = N1 + N2 - довжина програми,

n '= n1' + n2 '- теоретичний словник програми,

N '= n1 \* log2 (n1) + n2 \* log2 (n2) - теоретична довжина програми (для стилістично коректних програм відхилення N від N' не перевищує 10%)

V = N \* log2n - обсяг програми,

V '= N' \* log2n '- теоретичний об'єм програми, де n \* - теоретичний словник програми.

L = V '/ V - рівень якості програмування, для ідеальної програми L = 1

L '= (2 n2) / (n1 \* N2) - рівень якості програмування, для ідеальної програми L = 1

L '= (2 n2) / (n1 \* N2) - рівень якості програмування, заснований лише на параметрах реальної програми без урахування теоретичних параметрів,

EC = V / (L ') 2 - складність розуміння програми,

D = 1 / L '- трудомісткість кодування програми,

y '= V / D2 - рівень мови вираження

I = V / D - інформаційний зміст програми, дана характеристика дозволяє визначити розумові витрати на створення програми

E = N '\* log2 (n / L) - оцінка необхідних інтелектуальних зусиль при розробці програми, що характеризує число необхідних елементарних рішень при написанні програми

При застосуванні метрик Холстеда частково компенсуються недоліки, пов'язані з можливістю запису однієї і тієї ж функціональності різною кількістю рядків і операторів.

Ще одним типом метрик ПО, що відносяться до кількісних, є метрики Джілбі. Вони показують складність програмного забезпечення на основі насиченості програми умовними операторами або операторами циклу. Дана метрика, не дивлячись на свою простоту, досить добре відображає складність написання і розуміння програми, а при додаванні такого показника, як максимальний рівень вкладеності умовних і циклічних операторів, ефективність даної метрики значно зростає.

#### Робоче завдання

1. Відкрити створений програмний код адресної книги, що був згенерований на попередній роботі.
2. Порахувати кількісні метрики коду.
3. Провести аналіз метрик коду на прикладі свого коду.
4. Написати звіт по метрикам коду.

Контрольні питання:

1. Метрики складності потоку керування програми.
2. Метрики складності потоку керування даними.
3. Метрики складності потоку керування і даних програми.
4. Об'єктно-орієнтовані метрики.
5. Метрики надійності.
6. Гібридні метрики.
7. Loc- оцінки якості.
8. Метрики стилістики й зрозумілості програм
9. Метрики Холстеда.
10. Метрики циклічної складності за Мак-Кейбом.
11. Метрики Чепіна.