Міністерство освіти і науки України

Запорізький національний технічний університет

кафедра програмних засобів

**Реферат**

З дисципліни «Якість програмного забезпечення та тестування» з теми:

**«Метрика Пивоварського»**

Виконав:

студент групи КНТ-115 Ю.О. Лукашенко

Прийняла:

проф., д.т.н Г.В.Табунщик

2018

# ЗМІСТ

[ЗМІСТ 2](#_Toc528312103)

[ВСТУП 3](#_Toc528312104)

[1 МЕТРИКИ СКЛАДНОСТІ ПРОГРАМНИХ ПРОДУКТІВ 5](#_Toc528312105)

[1.1 Кількісні метрики 5](#_Toc528312106)

[1.2 Метрики складності потоку даних 6](#_Toc528312107)

[1.3 Метрики складності потоку керування програм 6](#_Toc528312108)

[1.4 Метрика Пивоварського 7](#_Toc528312109)

[ПЕРЕЛІК ПОСИЛАНЬ 8](#_Toc528312110)

# ВСТУП

Метрика Пивоварського є однією з метрик складності потоку керування програми, однак для розуміння її сутності необхідно спочатку з’ясувати базові поняття та визначення атрибутів програмних системи, що характеризують її якість – метрик якості.

**Метрика якості програмного забезпечення** є певною **моделлю** вимірювання сутності, що пов’язана з характеристикою якості програмного забезпечення, тобто є поєднанням конкретного методу вимірювання певного атрибуту та шкали, за якою проводиться це вимірювання[1]. За стандартом ISO/IEC 9126-2, вимірювати метрики можна у номінальній, порядковій, інтервальній, відносній та абсолютній шкалах[1,2].

Метрики якості представляють собою певну змінну, яку також називають **мірою**, при чому ця міра може бути **прямою**, якщо вона вимірюється безпосередньо, та **опосередкованою**, тобто такою, що залежить від результатів вимірювання інших метрик. В загальному випадку застосування метрик дозволяє визначити складність розробленого проекту, або проекту, що перебуває у розробці, оцінити об'єм робіт, стилістику розроблюваного проекту і зусилля, витрачені кожним розробником для реалізації того чи іншого рішення[3].

Основною метою використання метрик та їх застосування на практиці є оцінка якості не тільки якоїсь певної ділянки коду, а й розробки та усієї програмної системи в цілому. У такому контексті оцінка якості залежить також від таких понять, як критерії якості та характеристики якості. Зокрема, Д. МакКол запропонував такі критерії (показники) якості: функціональна повнота, точність обчислень, стійкість до помилок, модульність, захищеність, ефективність роботи тощо[1,4]. В свою чергу, характеристика якості програми – поняття, що відображає окремі фактори, що впливають на якість програм і піддаються вимірюванню.

Загалом, існуючі якісні оцінки програм можна виділити у такі групи[5]:

1. Метрики інформаційної складності програмних продуктів.
2. Метрики стабільності і надійності програмних систем.
3. Метрики архітектурної обґрунтованості і здатності системи до розширення.
4. Метрики якості написання коду та можливості його супроводження.
5. Метрики продуктивності праці програмістів для прогнозування термінів і строків розробки програм.

Детальніше розглянемо метрики інформаційної складності програмних продуктів, оскільки метрика Пивоварського відноситься до цієї групи.

# МЕТРИКИ СКЛАДНОСТІ ПРОГРАМНИХ ПРОДУКТІВ

При оцінці складності програм, як правило, виділяють три основні групи метрик [1,6]:

* кількісні метрики;
* метрики складності потоку даних програм;
* метрики складності потоку керування програм.

Розглянемо детальніше кожну групу та визначимо, до якої з них відноситься метрика Пивоварського.

## Кількісні метрики

Оцінки першої групи найбільш прості, тому набули широкого поширення. Найпоширенішою метрикою вихідного коду, що відображає розмір програмного проекту, є показник кількості рядків коду SLOC (Source Lines Of Code), при чому використовуються його два основні різновиди: кількість фізичних та логічних SLOC[6]. Кількість фізичних SLOC визначається як загальна кількість рядків коду, з урахуванням коментарів і порожніх рядків. Напротивагу фізичним, логічні метрики SI (Source Instructions) визначається як кількість команд і залежить від використовуваної мови програмування. Від метрики SLOC є багато похідних метрик, що дозволяють оцінити окремі показники проекту.

Безпосереднє вимірювання розміру програми, незважаючи на свою простоту, дає хороші результати. Звичайно, оцінка розміру програми недостатня для прийняття рішення про її складності, але цілком може бути застосована для класифікації програм, істотно розрізняються обсягами. При зменшенні відмінностей в обсязі програм на перший план висуваються оцінки інших факторів, що впливають на складність. Таким чином, оцінка розміру програми є оцінка за номінальною шкалою, на основі якої визначаються тільки категорії програм без уточнення оцінки для кожної категорії[1].

До кількісних метрик відносять також метрики Холстеда (взаємодія зі словниками операторів і операндів програми)[6,7] і метрики Джибла ­(складність програмного забезпечення на основі насиченості програми умовними операторами або операторами циклу).

## Метрики складності потоку даних

Інша група метрик складності програм – метрики складності потоку даних, тобто використання, конфігурації і розміщення даних в програмах.

До цих метрик відносять зокрема метрику Чепина (характеристика змінних програми за їхньою поведінкою), метрику Спена (відстань між першою і останньою появою змінної), метрику "модуль-глобальна змінна", а також міру Кафура, що створена на основі концепції локального і глобального інформаційних потоків тощо[1,6-7].

## Метрики складності потоку керування програм

Як правило, за допомогою цих метрик оперують поняттям керуючого графа програми, тобто щільністю керуючих переходів усередині програм, або взаємозв'язками цих переходів.

Традиційне уявлення програм у такому вигляді передбачає побудову керуючого орієнтованого графа G = (V, E), де V – вершини, відповідні послідовним обчисленням, а E – дуги, відповідні переходам від блоку до блоку.

Найпоширенішою метрикою, заснованою на аналізі отриманого графа, є цикломатична складність програми, зокрема, цикломатичне число Мак-Кейба, що визначається як V(G) = e – n + 2p, де e – кількість дуг, n – кількість вершин, p – кількість компонентів зв’язності[8]. 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 кількісті 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 цілому, метрики цикломатичної складності є досить гарним показником, що дозволяє своєчасно припинити подальше ускладнення окремих складових проекту і спростити їх, попередивши ймовірні проблеми з заплутаним і нестабільним кодом в майбутньому.

Крім того, від метрики цикломатичної складності з’явилось багато похідних метрик, зокрема і метрика Пивоварського.

## Метрика Пивоварського

Міра Пивоварського ставить за мету врахувати в оцінці складності програми відмінності не тільки між послідовними і вкладеними керуючими конструкціями, але і між структурованими і неструктурованими програмами.

Вона виражається відношенням:

де – модифікована цикломатична складність, обчислена так само, як і V(G), але з однією відмінністю: оператор CASE з n виходами розглядається як один логічний оператор, а не як n-1 операторів.

– глибина вкладеності i-тої предикатної вершини.

Для підрахунку глибини укладення предикатних вершин використовується число сфер впливу. Під глибиною вкладеності розуміється число всіх сфер впливу предикатів, які або повністю утримуватися в сфері даної вершини, або перетинаються з нею. Глибина вкладеності збільшується за рахунок вкладеності не самих предикатів, а «сфер впливу». Міра Пивоварського зростає при переході від послідовних програм до вкладених і далі до неструктурованих, що є її значною перевагою перед багатьма іншими метриками даної групи[1,9].

# ПЕРЕЛІК ПОСИЛАНЬ

1. Табунщик Г.В, Кудерметов Р.К., Каплієнко Т.І. / Т-12 Інженерія якості програмного забезпечення: навчальний посібник, 2-ге видання. – Запоріжжя: Дике Поле, 2016. – 176 с.
2. International Standard ISO/IEC 9126. Information technology – Software product evaluation – Quality characteristics and guidelines for their use. International Organization for Standardization, International Electrotechnical Commission, Geneva, 1991.
3. Метрика програмного забезпечення ­: [Електронний ресурс] ­­­– режим доступу: https://uk.wikipedia.org/wiki/Метрика\_програмного\_забезпечення
4. J. McCall, P. Richards, G. Walters. Factors in Software Quality. three volumes, NTIS AD-A049-014, AD-A049-015, AD-A049-055, November 1977.
5. Методи та засоби аналізу надійності функціонування програмного забезпечення з урахуванням етапів життєвого циклу ­: [Електронний ресурс] – http://old.lp.edu.ua/sites/default/files/dissertations/files/diss\_Yakovyna\_V\_S.pdf
6. Основные направления применения метрик : [Електронний ресурс] – режим доступу: http://csaa.ru/osnovnye-napravlenija-primenenija-metrik/
7. Программный код и его метрики : [Електронний ресурс] – режим доступy: https://habr.com/company/intel/blog/106082/
8. Цикломатична складність: [Електронний ресурс] – режим доступу: https://uk.wikipedia.org/wiki/Цикломатична\_складність
9. Метрики кода программного обеспечения: [Електронний ресурс] – режим доступу: https://www.viva64.com/ru/a/0045/