

Економіка ПЗ. Тема:

Оцінка характеристик програм на  
основі об'єктно-орієнтованих метрик

# Метрики Чидамбера і Кемерера

Автори запропонували 6 проектних метрик, основаних на аналізі методів класу, дерева наслідування і інших категоріях об'єктно-орієнтованого середовища.

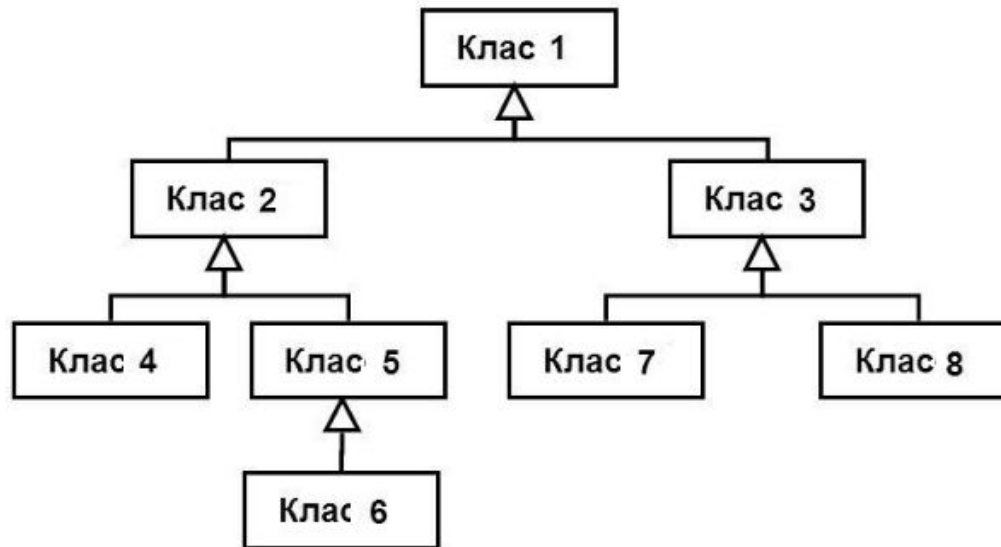
**1) Зважені методи на клас (WMC)** – дозволяє виміряти складність класів з врахуванням складності їх методів. Підраховувати кількість методів в класі достатньо складно.

Можливі два протилежних варіанти:

1. Підраховуються тільки методи поточного класу. Наслідовані методи ігноруються. Обґрунтування— наслідовані методи вже підраховані в тих класах, де вони визначились. Таким чином, інкрементність класу — кращий показник його функціональних можливостей, який відображає його право на існування.
2. Підраховуються методи, які визначені в поточному класі, і всі наслідовані методи. Цей підхід підкреслює важливість простору стану розуміння класу (а не інкрементності

# Метрики Чидамбера і Кемерера

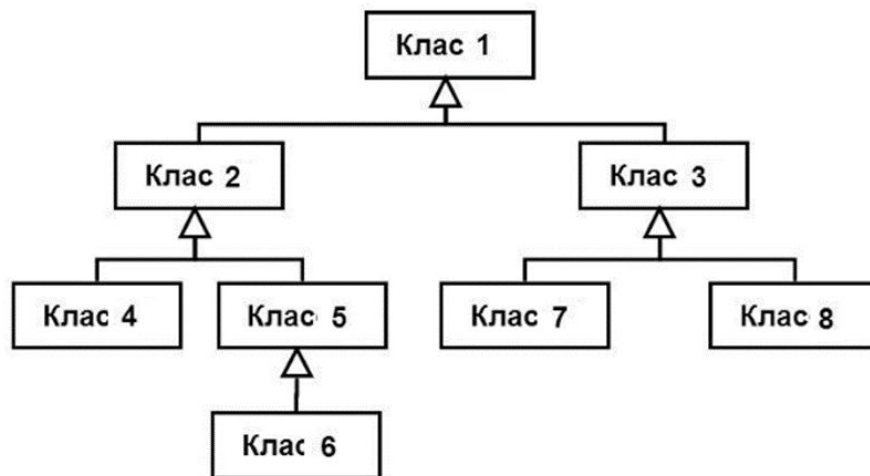
2. **Висота дерева наслідування (DIT)** – визначається як максимальна довжина шляху від листа до кореня дерева наслідування класів.



Відповідно, для окремого класу DIT, ця довжина максимального шляху даного класу до кореневого класу в ієрархії класів.

# Метрики Чидамбера і Кемерера

3. **Кількість нащадків (NOC)** – дозволяє визначити кількість безпосередніх нащадків даного класу.



Із збільшенням NOC зростає багаторазовість використання, так як наслідування — це форма повторного використання.

Метрики DIT і NOC — кількісні характеристики форми і розміру структури класів.

# Метрики Чидамбера і Кемерера

4. **Зв'язаність між класами об'єктів (СВО)** – надає можливість визначити кількість класів, з якими зв'язаний даний клас. Це має суттєве значення, коли один клас використовує методи або екземпляри іншого класу.

З'єднання означає, що методи даного класу використовують методи або екземплярні змінні іншого класу.



# Метрики Чидамбера і Кемерера

5. **Відгук для класа (RFC)** – дозволяє визначити кількість методів, яке може бути виконано у відповідь на отримання повідомлення даним класом. В цій метриці враховуються не тільки виконувані методи даного класу, але і методи інших класів, тобто за допомогою цієї метрики можна оцінити і ступінь потенційного «спілкування» конкретного класу з іншими класами: RFC- представляє собою кількість методів класу, збільшене на число методів інших класів, що викликаються з даного класу.



# Метрики Чидамбера і Кемерера

6. **Недолік зв'язаності в методах (LCOM)** – дозволяє оцінити залежність методів класу один від одного. Метрика показує наскільки методи не зв'язані один з одним через властивості.

Введемо позначення:

- НЕ ЗВ'ЯЗАНІ — кількість пар методів без загальних екземплярних змінних;
- ЗВ'ЯЗАНІ— кількість пар методів з загальними екземплярними змінними.
- $I_j$ — набір екземплярних змінних, які використовуються методом  $M_j$

$$LCOM = \begin{cases} \text{НЕ ЗВ'ЯЗАНІ} - \text{ЗВ'ЯЗАНІ, якщо НЕ ЗВ'ЯЗАНІ} > \text{ЗВ'ЯЗАНІХ} \\ 0 \text{ у протилежному випадку} \end{cases}$$

# Метрики Чидамбера і Кемерера

Приклад: В класі є методи:  $M1$ ,  $M2$ ,  $M3$ ,  $M4$ . Кожний метод працює зі своїм набором екземплярних змінних:

$$I_1 = \{a, b\}; I_2 = \{a, c\}; I_3 = \{x, y\}; I_4 = \{m, n\}.$$

Тоді:

$$\text{НЕ ЗВ'ЯЗАНІ} = \text{card}(I_{13}, I_{14}, I_{23}, I_{24}, I_{34}) = 5; \text{ЗВ'ЯЗАНІ} = \text{card}(I_{12}) = 1$$

$$LCOM = 5 - 1 = 4$$



# Метрики Чидамбера і Кемерера

## Контрольні завдання:

1) Дано 3 класи:

- А (включає приватні атрибути  $ra1$ ,  $ra2$  і методи  $opA1()$ ,  $opA2()$ )
- В (включає приватний атрибут  $rb1$ , і методи  $opB1()$ ,  $opB2()$ )
- С (включає приватні атрибути  $rc1$ ,  $rc2$  і методи  $opC1()$ ,  $opC2()$ )

Класи В і С наслідуються від класа А. Побудувати діаграму класів. Обчислити метрику WMC, DIT, NOSC.

# Метрики Чидамбера і Кемерера

2) Дано 4 класи:

- A (включає приватні атрибути pa1, pa2 і методи opA1(), opA2())

- B (включає приватний атрибут pb1, і методи opB1(), opB2())

- C (включає приватні атрибути pc1, pc2 і методи opC1(), opC2())

- D (включає приватні атрибути pd1, pd2 і методи opD1(), opD2())

Класи B, C наслідуються від класа A. Клас D від класа B. При виконанні операції opB1() визивається операція opD2(). Побудувати діаграму класів. Обчислити метрику СВО і RFC.

# Метрики Чидамбера і Кемерера

3) Дано 2 класи:

- А (включає приватні атрибути `pa1` типу `Integer`, `pa2` типу `Real`, `pa3` типу `Boolean` і методи `opA1(pa1: Integer)`, `opA2(pa2: Real)`, `opA3(opA1(pa1: Integer), pa3: Boolean)`)).

- В (включає приватний атрибут `pb1`, і методи `opB1()`, `opB2()`)

Клас В наслідуються від класа А. При виконанні операції `opB2()` визивається операція `opA1()`. Побудувати діаграму класів. Обчислити значення метрики WMC, CVO, RFC.

# Метрики Чидамбера і Кемерера

4) Визначити недолік зв'язаності LCOM

Клас
a, b, c, x, y, m, n
M1 {a, b} M2 {a, c} M3 {x, y} M4 {m, n}

# Метрики Лоренца і Кіда

1. Розмір класу (CS)
2. Кількість операцій, перевизначених підкласами (NOO)
3. Кількість операцій, доданих підкласами (NOA)
4. Індекс спеціалізації (SI)
5. Середній розмір операції (OSavg)
6. Складність операції (OC)
7. Середня кількість параметрів на операцію (NPavg)
8. Кількість описів сценаріїв (NSS)
9. Кількість ключевих класів (NKC)
10. Кількість підсистем (NSUB)

# Метрики Лоренца і Кіда

## 1. Розмір класу (CS)

Загальний розмір класу визначається за допомогою наступних вимірів:

Велике значення CS – клас має дуже багато обов'язків.

кількість інкапсульованих  
операцій класу

+

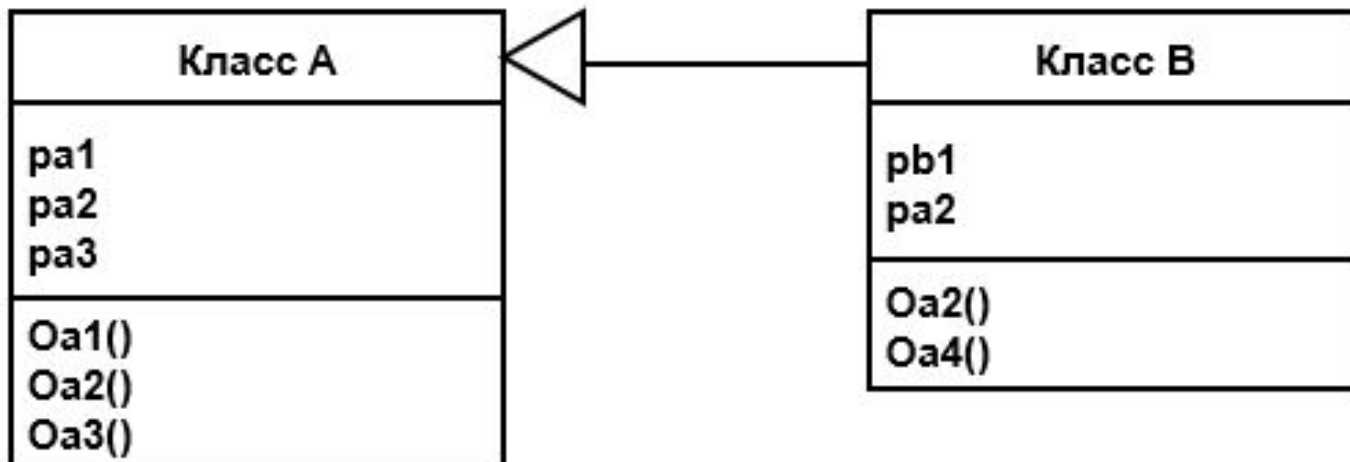
кількість інкапсульованих  
властивостей класу

# Метрики Лоренца і Кіда

## 2. Кількість операцій, перевизначених підкласом (NOO)

- Перевизначенням називають випадок, коли підклас заміщує операцію, наслідовану від головного класу, своєю власною версією.

Велик



□  $NOO(B) = 1$

# Метрики Лоренца і Кіда

## 3. Кількість операцій, доданих підкласом (NOA )

Підкласи спеціалізуються додаванням приватних операцій і властивостей.

З ростом NOA підклас знищується від абстракції головного класу.

Для рекомендованих значень  $CS = 20$  и  $DIT = 6$  рекомендоване значення NOA 4 метода.

## 4. Індекс спеціалізації (SI)

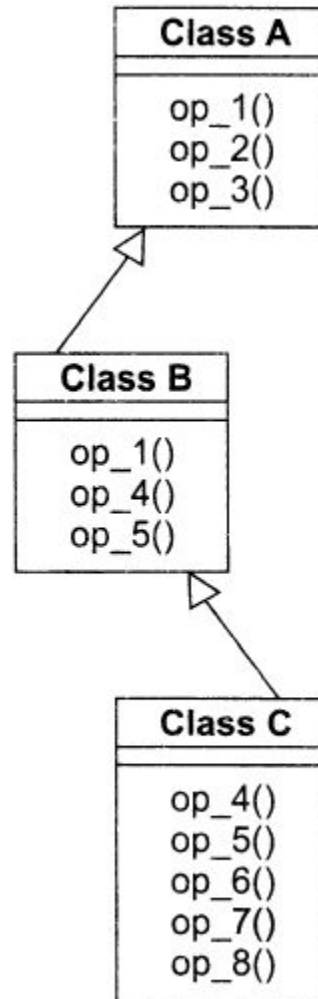
Забезпечує грубу оцінку степені спеціалізації кожного підкласу. Спеціалізація досягається додаванням, знищенням або перевизначенням операцій:

$$SI = (NOO \times \text{рівень}) / M_{\text{заг}}$$

где *рівень* — номер рівня в ієрархії, на якому знаходиться підклас,  $M_{\text{заг}}$  — загальна кількість методів класу.



# Метрики Лоренца і Кіда



# Метрики Лоренца і Кіда

## Операційно-орієнтовані метрики

### 5. Середній розмір операції ( $OS_{AVG}$ )

Як індикатор розміру може використовуватись кількість рядків програми, однак LOC-оцінки призводять до відомих проблем. Альтернативний варіант — «кількість повідомлень, посланих операцією».

### 6. Складність операції (ОС)

Складність операції може обчислюватись за допомогою стандартних метрик складності, тобто за допомогою LOC- або FP-оцінок, метрики цикломатичної складності, метрики Холстеда.

М. Лоренц і Д. Кідд пропонують обчислювати ОС сумуванням оцінок з ваговими коефіцієнтами

# Метрики Лоренца і Кіда

## Операційно-орієнтовані метрики

Параметр	Вес
Виклики функцій API	5,0
Присвоювання	0,5
Арифметичні операції	2,0
Повідомлення з параметрами	3,0
Вкладені вирази	0,5
Параметри	0,3
Прості виклики	7,0
Тимчасові змінні	0,5
Повідомлення без параметрів	1,0

# Метрики Лоренца і Кіда

## Операційно-орієнтовані метрики

### 7. Середня кількість параметрів на операцію $NP_{AVG}$

Чим більше параметрів у операції, тим складніше співпраця між об'єктами. Тому значення  $NP_{AVG}$  повинно бути як можна меншим.

Рекомендоване значення  $NP_{AVG} = 0,7$ .

## Метрики для ОО-проектів

### 8. Кількість описів сценаріїв (NSS)

Метрика NSS — ефективний індикатор розміру програми.

### 9. Кількість ключових класів (НКС)

Відображує об'єм розробки.

### 10. Кількість підсистем (NSUB)

Розміщення ресурсів, планування, загальні витрати на інтегрування.