

1. Metryki złożoności modułowej i międzymodułowej Chidamber & Kemerer (CK)

1.1. Podstawowe metryki CK:

- międzymodułowe CBO, RFC
- modułowe WMC, DIT, NOC, LCOM1.

1.2. Uzupełniony zbiór metryk przez innych autorów:

- międzymodułowe RFC', CA
- modułowe LCOM2, LCOM3, LCOM4, NPM

Instalacja programu CKJM

- **Link do strony z programem CKJM**

<http://www.spinellis.gr/sw/ckjm>

- **Ustawienie systemowej zmiennej środowiskowej Path:**

C:\PROGRA~1\Borland\CBUILD~1\Bin;C:\PROGRA~1\Borland\CBUILD~1\Projects\Bpl;%SystemRoot%\system32;%SystemRoot%;%SystemRoot%\System32\Wbem;c:\Program Files\Microsoft SQL Server\90\Tools\binn\; **c:\Program Files\apache-ant-1.7.1\bin\;**
c:\Program Files\apache-ant-1.7.1\lib\;

- **Warunki wywołania skryptu build.xml**

- Należy napisać skrypt build.xml wg wzoru podanego na następnych slajdach
- Należy umieścić skrypt build.xml w wybranym katalogu, w którym ma powstać plik z metrykami (np. ckjm.html podany w znaczniku <ckjm> oraz <xslt>)
- należy skopiować plik z rozszerzonym arkuszem stylu **ckjm_extra.xsl** z katalogu **ckjm-1.8\xsl** do wybranego katalogu
- Należy wywołać program **ant** w wybranym katalogu

Metrics information

Name	Range
V(G) (cyclomatic-complexity)	1.0 ... 10.0
LOC (total-lines-of-code)	5.0 ... 1000.0
NCLOC (noncomment-lines-of-code)	0.0 ... 0.0
CLOC (comment-lines-of-code)	0.0 ... 0.0
DC (density-of-comments)	0.2 ... 0.4
NP (number-of-parameters)	0.0 ... 4.0
EXEC (executable-statements)	0.0 ... 20.0
WMC (weighted-methods-per-class)	1.0 ... 50.0
RFC (response-for-class)	0.0 ... 50.0
DIT (depth-in-tree)	0.0 ... 5.0
NOC (number-of-children-in-tree)	0.0 ... 10.0
Ce (efferent-coupling)	0.0 ... 20.0
A (abstractness)	0.0 ... 0.5
NOT (number-of-types)	0.0 ... 80.0
NOTa (number-of-abstract-types)	0.0 ... 20.0
NOTc (number-of-concrete-types)	0.0 ... 80.0
NOTe (number-of-exported-types)	3.0 ... 50.0
LSP (limited-size-principle)	0.0 ... 10.0
DIP (dependency-inversion-principle)	0.3 ... 1.0
MQ (modularization-quality)	0.0 ... 1000.0
NT (number-of-tramps)	0.0 ... 1.0
LCOM (lack-of-cohesion)	0.0 ... 0.2

Zakresy wartości metryk, między innymi metryk CK

Przykład skryptu build.xml wywołanego domyślnie przez program ant dla programu typu Visual Web Java Server Faces

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="changeme" default="ckjm" basedir=".">
    <!-- wskazanie na plik ckjm-1.8.extended.jar: -->
    <property name="ckjm.location" value="c:/downloads/ckjm-1.8/ckjm-1.8.extended.jar" />
    <!-- to ładuje zadanie <ckjm> -->
    <taskdef name="ckjm" classname="gr.spinellis.ckjm.ant.CkjmTask">
        <classpath>
            <pathelement location="${ckjm.location}"/>
        </classpath>
    </taskdef>
    <!-- Metryki zostaną zapisane w pliku ckjm4.html, można również wybrać format xml.-->
    <target name="ckjm">
        <!-- ckjm lokalizuje pliki używane przez rozwijany projekt, przy założeniu, że 'build/classes' jest
        katalogiem, w którym są umieszczane 'bajtkody' (*.class)-->
        <ckjm outputfile="ckjm.xml" format="xml"
        classdir="e:/dydaktyka/d1/Wzorcowoprogramowania/WebWypożyczalnia3/build/web/WEB-INF/classes"/>
            <include name="**/*.class" />
            <exclude name="**/*Test.class" />
            <!-- ckjm lokalizuje w tej linii klasy wchodzące w skład rozwijanego projektu, przy założeniu, że
            build/classes' jest katalogiem w którym są umieszczane skompilowane klasy (*.class)-->
            <extdirs
            path="e:/dydaktyka/d1/Wzorcowoprogramowania/WebWypożyczalnia3/build/web/WEB-INF/classes"/>
                <!-- ckjm lokalizuje biblioteki używane przez rozwijany projekt, przy założeniu, że lib jest
                katalogiem, w którym są umieszczane biblioteki (*.jar)-->
                <extdirs path="e:/dydaktyka/d1/Wzorcowoprogramowania/WebWypożyczalnia3/build/web/WEB-INF/lib"/>
            </ckjm>
            <xslt in="ckjm.xml" style="ckjm_extra.xsl" out="ckjm1.html" />
        </target>
    </project>
```

Przykład aplikacji typu Visual Web
Java Server Faces - metryki CK

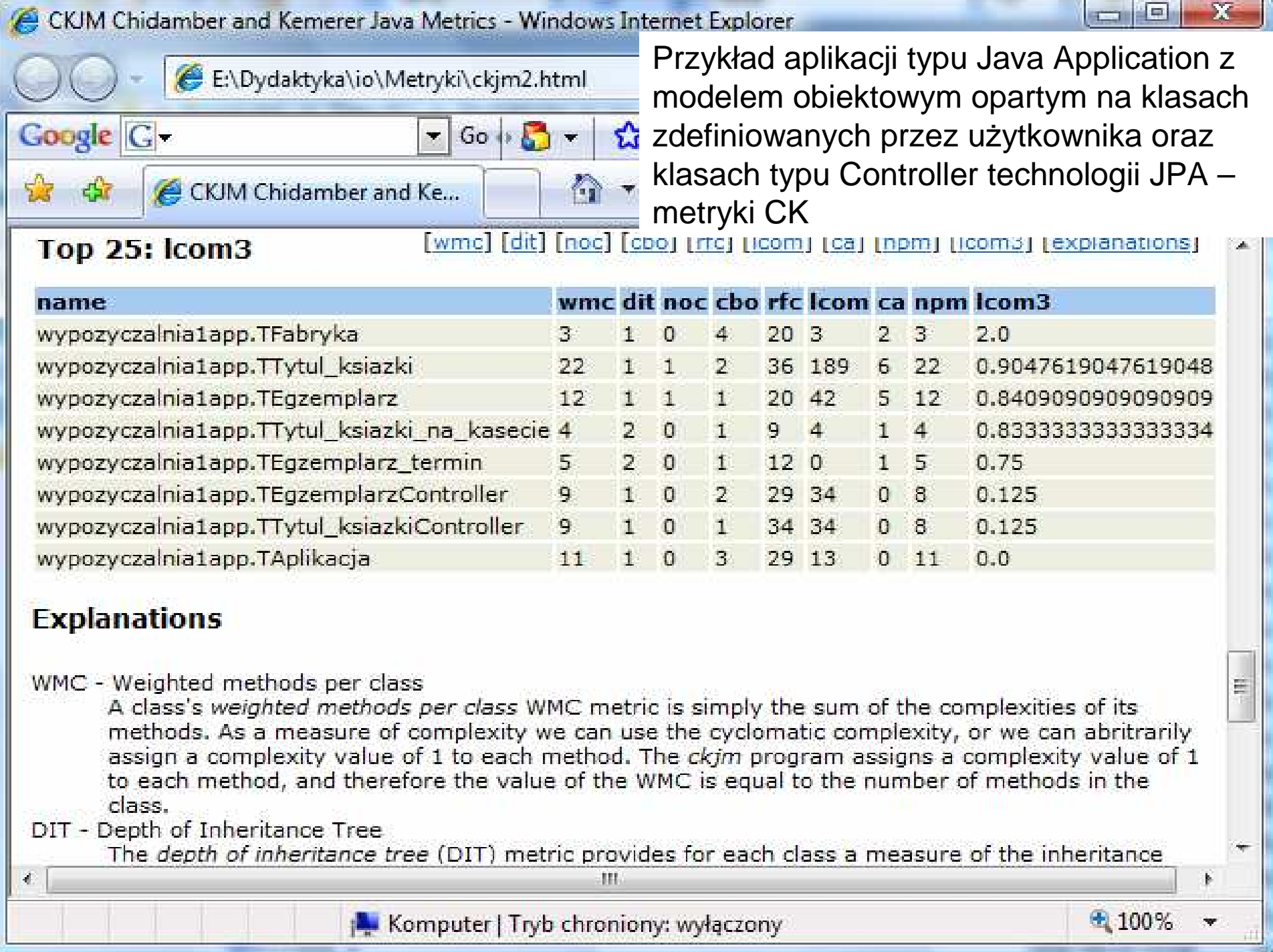
Top 25: lcom3

[\[wmc\]](#) [\[dit\]](#) [\[noc\]](#) [\[cbo\]](#) [\[rfc\]](#) [\[lcom\]](#) [\[ca\]](#) [\[npm\]](#) [\[lcom3\]](#) [\[explanations\]](#)

name	wmc	dit	noc	cbo	rfc	lcom	ca	npm	lcom3
webwypozyczalnia2.RequestBean1	6	3	0	3	11	15	14	3	2.0
webwypozyczalnia2.Tytulybaza	35	3	0	8	44	511	0	31	0.9411764705882353
webwypozyczalnia2.Ksiazkibaza	27	3	0	8	36	291	0	23	0.9230769230769231
webwypozyczalnia2.Tytuly	26	3	0	15	44	277	0	22	0.92
webwypozyczalnia2.Menu	26	3	0	5	32	283	6	22	0.92
webwypozyczalnia2.Baza_tytul	26	3	0	15	45	277	0	22	0.92
webwypozyczalnia2.Baza_ksiazki	24	3	0	13	40	234	0	20	0.9130434782608695
webwypozyczalnia2.Baza_tytuly	24	3	0	13	40	234	0	20	0.9130434782608695
webwypozyczalnia2.Ksiazki	24	3	0	18	49	234	0	20	0.9130434782608695
webwypozyczalnia2.Page1	22	3	0	12	40	195	0	18	0.9047619047619048
webwypozyczalnia2.FormTytul	31	3	0	6	43	349	2	27	0.9
webwypozyczalnia2.FormKsiazka	19	3	0	6	31	115	1	15	0.8333333333333334
webwypozyczalnia2.Logo	11	3	0	6	18	43	0	7	0.8
webwypozyczalnia2.ApplicationBean1	25	3	0	7	52	234	16	24	0.7666666666666666
webwypozyczalnia2.SessionBean1	9	3	0	2	14	30	15	7	0.75
webwypozyczalnia2.Ksiazkiaplikacja	9	3	0	5	15	30	0	5	0.75
webwypozyczalnia2.Tytulyaplikacja	11	3	0	8	25	43	1	7	0.6

Przykład skryptu build.xml wywołanego domyślnie przez program ant dla programu typu Java Application

```
<?xml version="1.0" encoding="UTF-8"?>
<project name="changeme" default="ckjm" basedir=". ">
    <!-- wskazanie na plik ckjm-1.8.extended.jar: -->
    <property name="ckjm.location" value="c:/downloads/ckjm-1.8/ckjm-1.8.extended.jar" />
        <!-- to laduje zadanie <ckjm> -->
    <taskdef name="ckjm" classname="gr.spinellis.ckjm.ant.CkjmTask">
        <classpath>
            <pathelement location="{ckjm.location}"/>
        </classpath>
    </taskdef>
        <!-- Metryki zostana zapisane w pliku ckjm4.html, mozna rowniez wybrac format xml.-->
    <target name="ckjm">
        <!-- ckjm lokalizuje pliki uzywane przez rozwijany projekt, przy zalozeniu, ze 'build/classes' jest katalogiem, w ktorym sa umieszczane 'bajtkody' (*.class)-->
        <ckjm outputfile="ckjm.xml" format="xml,,
            classdir="e:/dydaktyka/d1/Wzorceoprogramowania/Wypozyczalnia1app/build/classes/">
            <include name="**/*.class" />
            <exclude name="**/*Test.class" />
                <!-- ckjm lokalizuje w tej linii klasy wchodzace w sklad rozwijanego projektu, przy zalozeniu, ze build/classes' jest katalogiem w ktorym sa umieszczane skompilowane klasy (*.class)-->
            <extdirs path="e:/dydaktyka/d1/Wzorceoprogramowania/Wypozyczalnia1app/build/classes/" />
                <!-- ckjm lokalizuje biblioteki uzywane przez rozwijany projekt, przy zalozeniu, ze lib jest katalogiem, w ktorym sa umieszczane biblioteki (*.jar)-->
            <extdirs path="e:/dydaktyka/d1/Wzorceoprogramowania/Wypozyczalnia1app/dist/lib"/>
        </ckjm>
        <xslt in="ckjm.xml" style="ckjm_extra.xsl" out="ckjm2.html" />
    </target>
</project>
```



Przykład aplikacji typu Java Application z modelem obiektowym opartym na klasach zdefiniowanych przez użytkownika oraz klasach typu Controller technologii JPA – metryki CK

Top 25: lcom3

[\[wmc\]](#) [\[dit\]](#) [\[noc\]](#) [\[cbo\]](#) [\[rfc\]](#) [\[lcom\]](#) [\[ca\]](#) [\[npm\]](#) [\[lcom3\]](#) [\[explanations\]](#)

name	wmc	dit	noc	cbo	rfc	lcom	ca	npm	lcom3
wypozyczalnia1app.TFabryka	3	1	0	4	20	3	2	3	2.0
wypozyczalnia1app.TTytul_książki	22	1	1	2	36	189	6	22	0.9047619047619048
wypozyczalnia1app.TEgzemplarz	12	1	1	1	20	42	5	12	0.8409090909090909
wypozyczalnia1app.TTytul_książki_na_kasecie	4	2	0	1	9	4	1	4	0.8333333333333334
wypozyczalnia1app.TEgzemplarz_termin	5	2	0	1	12	0	1	5	0.75
wypozyczalnia1app.TEgzemplarzController	9	1	0	2	29	34	0	8	0.125
wypozyczalnia1app.TTytul_książkiController	9	1	0	1	34	34	0	8	0.125
wypozyczalnia1app.TAplikacja	11	1	0	3	29	13	0	11	0.0

Explanations

WMC - Weighted methods per class
A class's *weighted methods per class* WMC metric is simply the sum of the complexities of its methods. As a measure of complexity we can use the cyclomatic complexity, or we can arbitrarily assign a complexity value of 1 to each method. The *ckjm* program assigns a complexity value of 1 to each method, and therefore the value of the WMC is equal to the number of methods in the class.

DIT - Depth of Inheritance Tree
The *depth of inheritance tree* (DIT) metric provides for each class a measure of the inheritance