Measuring Object Oriented Structure

for

Software Metrics Lab

Prepared by

Naimur Rahman (ASH1925011M)

Sunaan Sultan (ASH1925004M)

Arnab Dey (ASH1925024M)

Institute of Information Technology

Noakhali Science and Technology University

5th March, 202

.

Coupling Between Objects

CBO specifically measures the number of other classes that a particular class depends on or is coupled to. In other words, it measures the number of other classes that a class uses or references in its methods, attributes, or relationships. A high CBO value indicates that a class has a high degree of coupling with other classes in the system, which can make the system more complex and difficult to modify or maintain

Classes	Value of CBO
AuthenticationActivityTest	1
ExampleInstrumentedTest	0
AllCourseDetailsActivity	3
AuthenticationActivity	2
CourseCoordinatorActivity	3
DevelopersActivity	3
DirectorProfileActivity	3
HomeActivity	13
IntroActivity	1
MainActivity	2
NoticeBoardActivity	5
OfficialsActivity	3
ProfileActivity	2
RegisterUserActivity	2
SearchActivity	8
StudentAdapterActivity	3
StudentInfoActivity	3
SyllabusActivity	2
TeachersInfoActivity	3
ExampleUnitTest	0
BatchCard	4
Card	0
CourseCoCard	4
NoticeCard	4
OfficialsCard	4
StudentCard	5
TeacherCard	4
Batch	0
CourseCo	0
Notice	0
Official	0
Student	0
Teacher	0
LoadingDialog	1
PdfViewer	3

.

Lack of Cohesion in Methods (LCOM)

LCOM = the number of non-cohesive pairs of methods

Classes	Value of LCOM
AuthenticationActivityTest	1
ExampleInstrumentedTest	1
AllCourseDetailsActivity	1
AuthenticationActivity	1
CourseCoordinatorActivity	1
DevelopersActivity	1
DirectorProfileActivity	1
HomeActivity	13
IntroActivity	1
MainActivity	1
NoticeBoardActivity	1
OfficialsActivity	1
ProfileActivity	1
RegisterUserActivity	2
SearchActivity	2
StudentAdapterActivity	1
StudentInfoActivity	1
SyllabusActivity	9
TeachersInfoActivity	1
ExampleUnitTest	1
BatchCard	0
Card	0
CourseCoCard	0
NoticeCard	0
OfficialsCard	0
StudentCard	0
TeacherCard	0
Batch	0
CourseCo	0
Notice	1
Official	0
Student	0
Teacher	0
LoadingDialog	1
PdfViewer	1

.

Response for Class RFC

RFC = count the number of unique methods that can be called in response to a message to an object of that class

Classes	Value of RFC
AuthenticationActivityTest	2
ExampleInstrumentedTest	1
AllCourseDetailsActivity	4
AuthenticationActivity	1
CourseCoordinatorActivity	4
DevelopersActivity	1
DirectorProfileActivity	4
HomeActivity	13
IntroActivity	1
MainActivity	1
NoticeBoardActivity	7
OfficialsActivity	4
ProfileActivity	1
RegisterUserActivity	5
SearchActivity	21
StudentAdapterActivity	4
StudentInfoActivity	4
SyllabusActivity	9
TeachersInfoActivity	4
ExampleUnitTest	1
BatchCard	7
Card	2
CourseCoCard	10
NoticeCard	7
OfficialsCard	9
StudentCard	9
TeacherCard	9
Batch	9
CourseCo	13
Notice	10
Official	11
Student	6
Teacher	11
LoadingDialog	3
PdfViewer	4

Depth of Inheritance Tree (DIT)

Depth of Inheritance Tree (DIT) is a metric used in Object-Oriented Structure Measurement (OOSM) to measure the number of levels in the inheritance tree for a class. A high DIT value indicates that a class is at a deeper level in the inheritance tree and may be more difficult to understand and maintain.

Classes	Value of DIT
AuthenticationActivityTest	0
ExampleInstrumentedTest	0
AllCourseDetailsActivity	0
AuthenticationActivity	0
CourseCoordinatorActivity	0
DevelopersActivity	0
DirectorProfileActivity	0
HomeActivity	0
IntroActivity	0
MainActivity	0
NoticeBoardActivity	0
OfficialsActivity	0
ProfileActivity	0
RegisterUserActivity	0
SearchActivity	0
StudentAdapterActivity	0
StudentInfoActivity	0
SyllabusActivity	0
TeachersInfoActivity	0
ExampleUnitTest	0
BatchCard	1
Card	0
CourseCoCard	1
NoticeCard	1
OfficialsCard	1
StudentCard	1
TeacherCard	1
Batch	0
CourseCo	0
Notice	0
Official	0
Student	0
Teacher	0
LoadingDialog	0
PdfViewer	0

Number Of Children (NOC)

Number of Children (NOC) is a metric to measure the number of immediate subclasses of a class. A high NOC value indicates that a class has many immediate subclasses

Classes	Value of NOC
AuthenticationActivityTest	0
ExampleInstrumentedTest	0
AllCourseDetailsActivity	0
AuthenticationActivity	0
CourseCoordinatorActivity	0
DevelopersActivity	0
DirectorProfileActivity	0
HomeActivity	0
IntroActivity	0
MainActivity	0
NoticeBoardActivity	0
OfficialsActivity	0
ProfileActivity	0
RegisterUserActivity	0
SearchActivity	0
StudentAdapterActivity	0
StudentInfoActivity	0
SyllabusActivity	0
TeachersInfoActivity	0
ExampleUnitTest	0
BatchCard	0
Card	6
CourseCoCard	0
NoticeCard	0
OfficialsCard	0
StudentCard	0
TeacherCard	0
Batch	0
CourseCo	0
Notice	0
Official	0
Student	0
Teacher	0
LoadingDialog	0
PdfViewer	0

.

Nested Loop Matrix NLM

NLM stands for Nested Loop Metrics, which is a set of software metrics used to measure the complexity of nested loops in code.

Classes	Value of NLM
AuthenticationActivityTest	2
ExampleInstrumentedTest	1
AllCourseDetailsActivity	2
AuthenticationActivity	1
CourseCoordinatorActivity	2
DevelopersActivity	1
DirectorProfileActivity	2
HomeActivity	13
IntroActivity	1
MainActivity	1
NoticeBoardActivity	4
OfficialsActivity	2
ProfileActivity	1
RegisterUserActivity	5
SearchActivity	8
StudentAdapterActivity	2
StudentInfoActivity	2
SyllabusActivity	9
TeachersInfoActivity	2
ExampleUnitTest	1
BatchCard	3
Card	2
CourseCoCard	3
NoticeCard	3
OfficialsCard	3
StudentCard	3
TeacherCard	3
Batch	9
CourseCo	13
Notice	10
Official	11
Student	6
Teacher	11
LoadingDialog	3
PdfViewer	2

.

Nesting Metrics

NM stands for Nesting Metrics, which is a set of software metrics used to measure the complexity of control structures in code.

There are three main NM metrics:

NM1: Number of Conditional Statements
NM2: Number of Looping Statements
NM3: Number of Control Variables

Classes	Value of NM
AuthenticationActivityTest	2
ExampleInstrumentedTest	1
AllCourseDetailsActivity	2
AuthenticationActivity	1
CourseCoordinatorActivity	2
DevelopersActivity	1
DirectorProfileActivity	2
HomeActivity	13
IntroActivity	1
MainActivity	1
NoticeBoardActivity	4
OfficialsActivity	2
ProfileActivity	1
RegisterUserActivity	5
SearchActivity	8
StudentAdapterActivity	2
StudentInfoActivity	2
SyllabusActivity	9
TeachersInfoActivity	2
ExampleUnitTest	1
BatchCard	4
Card	2
CourseCoCard	4
NoticeCard	4
OfficialsCard	4
StudentCard	4
TeacherCard	4
Batch	9
CourseCo	13
Notice	10
Official	11
Student	6
Teacher	11
LoadingDialog	3
PdfViewer	2

.

Total Number Of Methods

TNM stands for Total Number of Methods, which is a software metric used to measure the size and complexity of a software system. The TNM metric is calculated by summing the number of methods in all the classes of a software system.

Classes	Value of NM
AuthenticationActivityTest	4
ExampleInstrumentedTest	1
AllCourseDetailsActivity	2
AuthenticationActivity	2
CourseCoordinatorActivity	2
DevelopersActivity	1
DirectorProfileActivity	2
HomeActivity	15
IntroActivity	1
MainActivity	1
NoticeBoardActivity	4
OfficialsActivity	2
ProfileActivity	1
RegisterUserActivity	6
SearchActivity	11
StudentAdapterActivity	2
StudentInfoActivity	2
SyllabusActivity	9
TeachersInfoActivity	2
ExampleUnitTest	1
BatchCard	4
Card	2
CourseCoCard	4
NoticeCard	4
OfficialsCard	4
StudentCard	4
TeacherCard	4
Batch	9
CourseCo	13
Notice	10
Official	11
Student	6
Teacher	11
LoadingDialog	3
PdfViewer	2