OBJECT-ORIENTED MODELLING & PROGRAMMING

SCS 1310

Lasanthi De Silva & Viraj Welgama

Lnc@ucsc.cmb.ac.lk; www@ucsc.cmb.ac.lk

2

WHAT DO YOU WANT TO LEARN IN THIS COURSE…

COURSE INTRODUCTION

COURSE SPECIFICS

NO OF CREDITS: 3
(2L + 1P)

SEMESTER: SECOND

CORE / OPTIONAL:
CORE

LECTURES:
2 HRS
(TOTAL: 30 HRS)

LAB SESSIONS:
2 HRS
(TOTAL: 30 HRS)

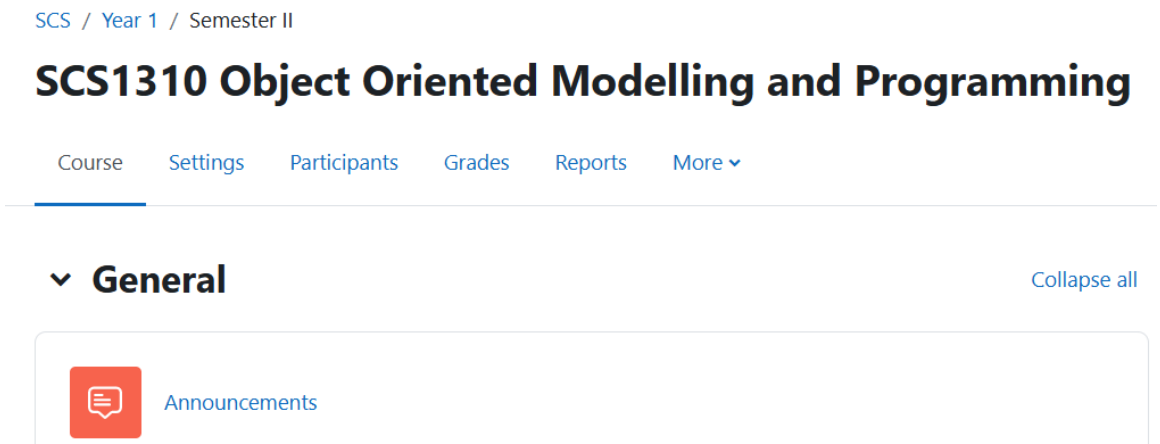
EVALUATION CRITERIA

- Assignments: 30%
  - Lab Session Submissions
  - In class Assignments/ Take Home Assignment
- Final Exam: 70%
  - 20 MCQs (40 Marks) and 2 Structured Questions (30 Marks each)

PASS MARK: 40

LECTURES & LAB SESSIONS

- Lectures: Thursdays from 10.00 am to 12.00 pm
- Lab Sessions
  - Lab Session (Gr1 & Gr2)
  - Day & Time: Monday (Gp 2) & Friday (Gp 1)
- Course Notifications: via UGVLE



COURSE DESCRIPTION

This course provides a comprehensive introduction to the **Object-Oriented (OO) paradigm**, emphasizing the **design** and **implementation** of applications using OO principles.

By the end of the course, students will gain a solid understanding of **OOP concepts and methodologies**, enabling them to design and write **efficient, scalable, and maintainable code**.

The course lays a strong foundation for students to enhance their sw **design/programming skills** and pursue successful **careers in software development**.

INTENDED LEARNING OUTCOMES

By the end of this course, students will be able to:

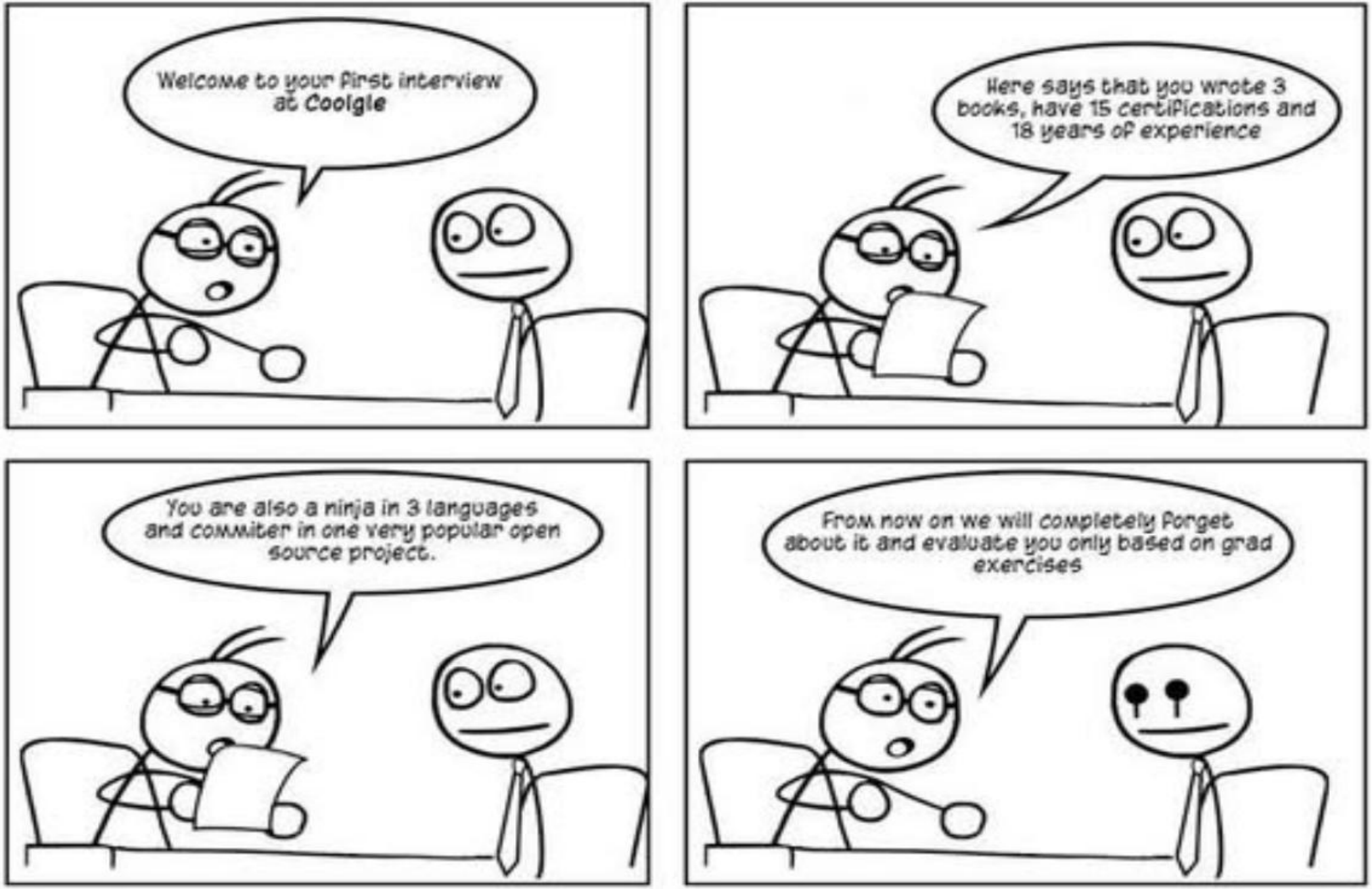
- Explain key **OOP concepts** like classes, objects, methods, properties, and relationships.
- Utilize **abstraction, encapsulation, polymorphism**, and **inheritance** in program design.
- Create and interpret **UML diagrams** such as use case, class, sequence diagrams etc.
- Design **reusable, flexible, and maintainable** code using best practices.
- Develop robust **error-handling mechanisms** with custom exception classes.
- Apply function and **class templates** for generic programming.

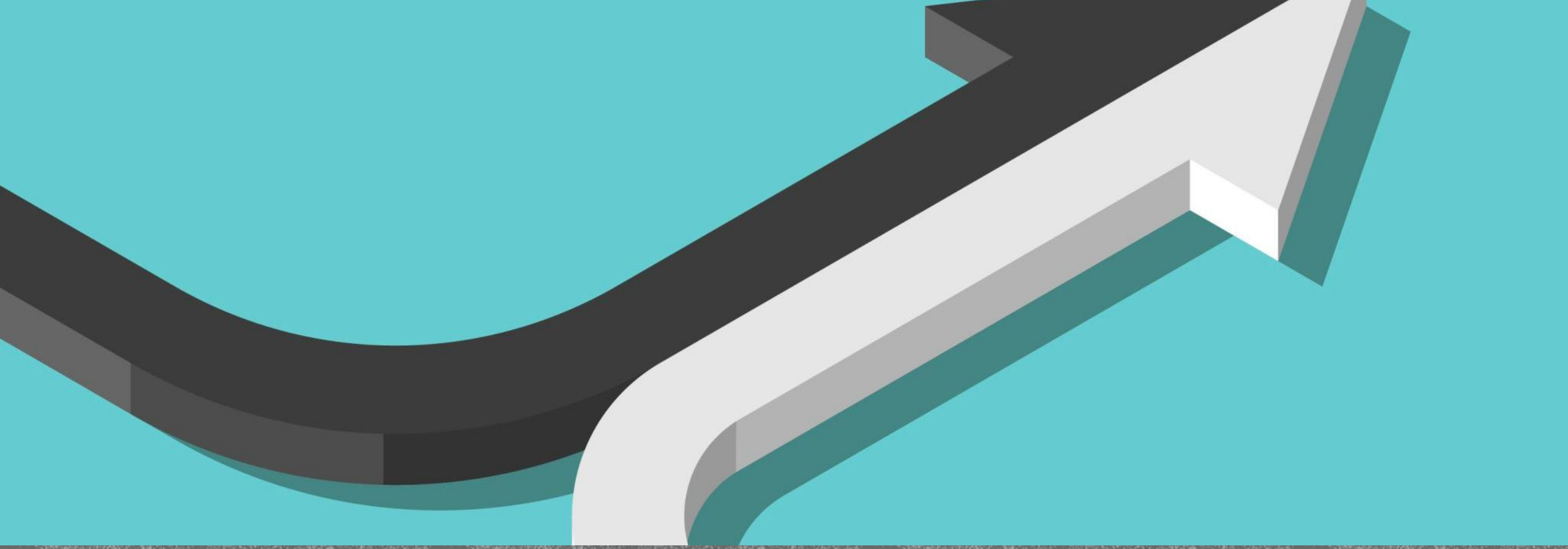
COURSE CONTENT

- Object Oriented Paradigm (4 Weeks – LNC)
- Fundamental Object-Oriented Concepts (4 Weeks – WVW)
- Unified Modelling Language (4 Weeks – LNC)
- Code Reusability and Flexibility (1 Week – WVW)
- Exception Handling (1 Weeks- WVW)
- Templates in C++ (1 Weeks- WVW)

COURSE REFERENCES

- Robert Lafore, 4th Edition, Object Oriented Programming in C++.
- Balagurusamy, E., 2021. Object-oriented programming with C++.
- Martin, R.C., 2009. Clean code: a handbook of agile software craftsmanship. Pearson Education.
- Booch, G., Maksimchuk, R.A., Engle, M.W., Young, B.J., Connallen, J. and Houston, K.A., 2008. Object-oriented analysis and design with applications. Addison-Wesley.


Welcome to your first interview at Coolgle
Here says that you wrote 3 books, have 15 certifications and 18 years of experience
You are also a ninja in 3 languages and commiter in one very popular open source project.
From now on we will completely forget about it and evaluate you only based on grad exercises



NEXT...

Object Oriented Paradigm