

# COMP319 REVISION

# Exam format

- Part A Answer all questions
  - 70 Marks
- Part B
  - Answer 1 out of 2

# Topics OO development

- Open/closed principle in OO development
  - Explain the term open/closed
  - Explain the benefits
  - Show using code example how open closed is implemented using
    - interface definition (interface closure)
    - final modification on public methods
    - virtual protected methods

# Factory and abstract factory class

- Class diagram
- Example in real life
- Why we bother?
  - Benefits

# Topics OO patterns

- Model view controller (MVC)
  - Explain
    - How it works
    - What benefits are provided by
    - Give example of how MVC operation would happen in practice (example something like a web email application)

# Other patterns to revise

- Singleton
- Chain of responsibility
- Memento
- Double lock checking

# AOP

- Why we need AOP
  - Issue of cross cut concerns and code cloning, scattering and tangling
- How AOP works
- Concept of
  - An aspect
  - Advice code
  - Pointcuts and joinpoints
  - Brief example

# Program slicing

- Be able to produce a
  - Dependency graph for a small piece of code <8 lines of code
- Be able to do both (use tree)
  - Forward slicing
    - Start with variable assignment
      - Look for all statements effected and include in slice
  - Backward slicing
    - Look for all statements which effect output at given line

# Program slicing

- Application of
  - Backward slicing
  - Why do we use it?
  - Forward slicing
  - Why do we use it?

# Project scheduling

- How to manage a schedule
- Dealing with slippage
- Strategies' for handling slippage
  - Re-scheduling
  - Re-resourcing
  - Using triage
    - See hand outs for detail

# Cost estimation and project failure

- Rise and fall of Chaos report
  - Problems with Chaos report
  - Boem's cone of uncertainty
  - Using f/a as an indication of project success
  - Results of different case studies examined
- EQF
  - What is it?
  - How is it calculated?
  - Why is it important?

# Threading

- Threads and tasks
  - Difference
- Issues with shared data
- Deadlock (what is it, how it happens)
- Lost updates (what are they, how do they happen)
- Thread starvation
- Testing for deadlock and race condition issues

# Actor model

- All aspects of the Actor model
  - What is it?
    - Full description?
  - Mobility?
    - Strong and weak
  - Location transparency?
    - How is this handled?

# AOP

- Cross cutting concerns
- Description with example of (in AspectJ)
  - Jointpoints
  - Pointcuts
  - Aspect
  - Advice
- Possible applications of AOP