Distributed Systems and Security Revision Lecture

Brad Karp

UCL Computer Science

CS M030 / GZ03

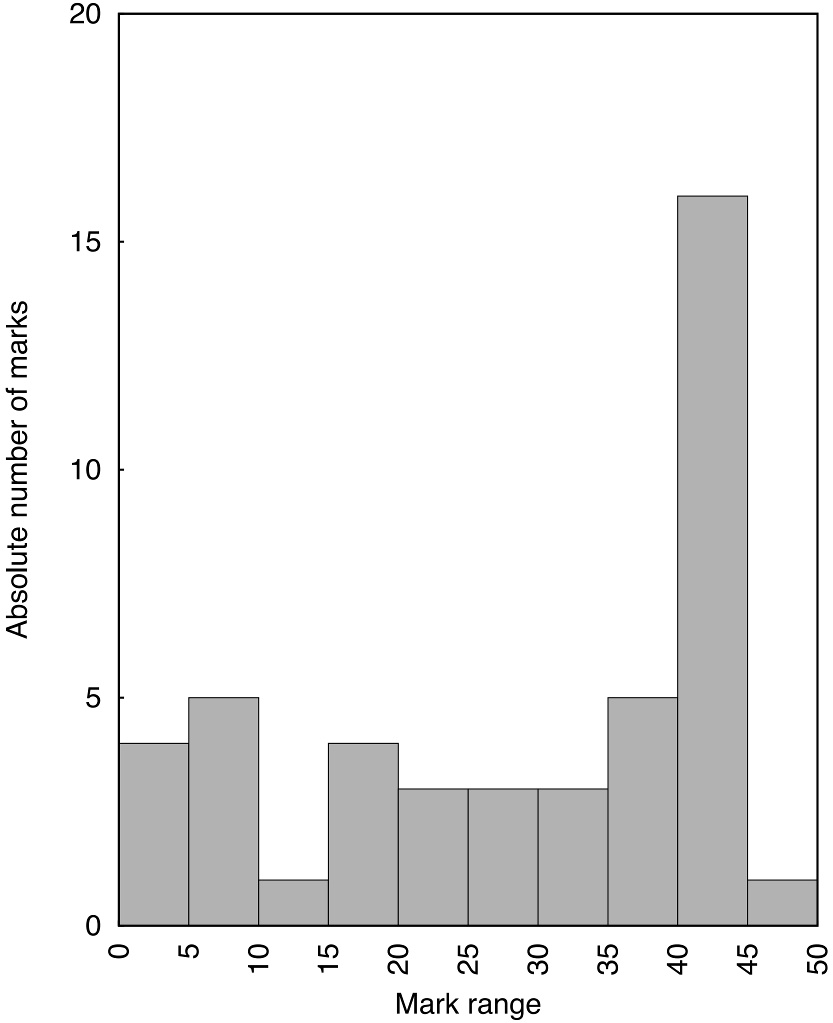
27th April, 2010

Marking Scheme

- Coursework 1: 10%
- Coursework 2: 5%
- Final exam: 85%

- Must average >= 50% across all to pass

Coursework 2 Results



Exam Rubric

- Four multi-part short-answer questions
  - you must answer two of your choice
  - questions on past papers for M030/GZ03 are representative
- One set of multiple-choice questions that you must answer
  - new this year
- Each question worth 33 marks

Multiple-Choice Questions

- A scenario followed by up to five lettered statements
- You must fill in **only** the letters of the correct statements
- Answers go on multiple-choice answer sheet, must be written in HB pencil
- Anywhere between zero and all five lettered statements may be true
- Marking scheme:
  - Gain a mark for each correct answer filled in
  - Lose a mark for each incorrect answer filled in
  - Lowest total multiple-choice mark possible is zero
  - Marks normalized into [0, 33]

Exam Style

- Questions will test understanding of concepts and system designs taught in lecture
  - Not just memorization, but why and how the systems work, and when they don’t
- Questions will ask you to apply knowledge to solve problems you haven’t seen before

What We Won’t Ask You

- “Write out this pseudocode from memory.”
  - No point, doesn’t test how well you think, nor how well you’ve mastered ability to apply material!
- “Explain how System X will behave in this utterly bizarre corner case that is insignificant in reality, and was never mentioned in lecture.”
  - We’re trying to determine whether you understand central themes in the papers, and how to apply the ideas in the papers.

What We Might Ask You

- “How does this system behave in this common situation?”
- “What happens when you make this change to this system’s design?”
- “Why does this system do this task in this way?”

- Questions may span multiple topics!

What We Might Ask You

- "How does this system behave in this common situation?"
- "What happens when you make this change to this system's design?"

Substantiate your answers! The right conclusion must be supported by relevant details.

- Questions may span multiple topics!

How Should You Study?

- **Re-read lecture notes, re-read papers**
- For distributed systems topics, be sure you understand full details of examples worked through in lecture
  - Try perturbing an example; see if you can solve perturbed form
  - Understand why each part of algorithm/system needed
- Accumulate your questions on material, discuss in group
  - Surprisingly time-efficient
  - Pool your understanding; likely >= one of you understands each aspect of each topic
  - Only works if you prepare by re-reading lecture notes and working through examples first!