COMP9020
Term 3, 2019
Course Review

Course Review

Goal: for you to become a competent computer **scientist**.

Requires an understanding of fundamental concepts:

- Number Theory, Sets, Relations and Functions
- Recursion, Induction and Asymptotic Analysis
- Logic
- Graph Theory
- Combinatorics and Probability

In CS/CE these are used to:

- formalise problem specifications and requirements
- develop abstract solutions (algorithms)
- analyse and prove properties of your programs

Assessment Summary

Quizzes: Best 10 out of 16 – (10 marks)
Assignments: 3 assignments worth 10 marks each – (30 marks)
Final exam – (60 marks)

NB

You must achieve 40% on the final exam AND 50% overall to pass the course.

Final Exam

Goal: to check whether you are a competent computer scientist.

Requires you to demonstrate:

- understanding of mathematical concepts
- ability to apply these concepts and explain how they work

Lectures, quizzes and assignments have built you up to this point.

Final Exam

Tuesday, 3 December, 8:45am
Kensington Room, Randwick Racecourse

- 10 short-answer questions plus 5 long-answer questions
- Covers **all** of the contents of this course.
- Each short-answer question is worth 3 marks ($10 \times 3 = 30$)
  Each long-answer question is worth 18 marks ($5 \times 18 = 90$)
  Total exam marks $= 120$ (i.e. 1 mark/minute)
- Time allowed – 120 minutes + 10 minutes reading time
- *Closed book*. One handwritten or typed A4-sized sheet (double-sided is ok) of your own notes
- Three answer booklets – one for rough work and two for answers. Clearly label each booklet.

Exam questions

Short answer questions (3 marks each)

- No justification necessary, but partial marks can be awarded for incorrect answers that demonstrate a level of understanding.
- No negative marking
- Don't have to answer each on its own page

Long answer questions (18 marks each)

- **Start each question on a new page**
- **List the questions attempted on the front of the booklet** in the order they were attempted
- Unless specified, *any* valid proof technique is acceptable

NB

Questions are culturally neutral – no concepts other than those taught in this course are assumed.

Revision Strategy

- Re-read lecture slides
- Read the corresponding chapters in the book (R & W)
- **Review/solve assignments and quizzes**
- Review/solve problem sets
- Solve more problems from the book
- Practice multiple choice questions

(Applying mathematical concepts to solve problems is a skill that improves with practice)

Additional consultation times:

- Tuesday November 26 (online) 9PM
- Wednesday November 27 (Rm 204, K17) 2-3PM

Supplementary Exam

You can apply formally for special consideration

- a supplementary examination may or may not be granted
- a supplementary examination is typically more difficult than the original examination

If you attend an exam

- you are making a statement that you are "fit and healthy enough"
- it is your only chance to pass (i.e. no second chances)

If your overall result is $\geq$ **47** you can sit the supplementary exam, in which you must score 50 or higher to pass

Assessment

Assessment is about determining how well you understand the syllabus of this course.

If you can't demonstrate your understanding, you don't pass.

In particular, I can't pass people just because ...

- please, please, ... my family/friends will be ashamed of me
- please, please, ... I tried really hard in this course
- please, please, ... I'll be excluded if I fail COMP9020
- please, please, ... this is my final course to graduate
- etc. etc. etc.

(Failure is a fact of life. For example, my scientific papers or project proposals get rejected sometimes too)

Assessment (cont'd)

Of course, assessment isn't a "one-way street" ...

- I get to assess you in the final exam
- you get to assess me in UNSW's MyExperience Evaluation
  - go to `https://myexperience.unsw.edu.au/`
  - login using zID@ad.unsw.edu.au and your zPass

Please fill it out ...

- give me some feedback on how you might like the course to run in the future
- even if that is "Exactly the same. It was perfect this time."

Content review

Week 1: Proof techniques and Number Theory

- Proof strategies
- Number theory
  - Floor $\lfloor \cdot \rfloor$ and ceiling $\lceil \cdot \rceil$ functions
  - Interval $[m, n]$, (m, n), etc
  - Absolute value $|\cdot|$ function
  - Divides relation $m|n$
  - Greatest common divisor, least common multiple
  - div, %, mod

Week 1: Need to know

- How to present proofs
- Number theory definitions and notation. Especially:
  - Interval notation
  - Divides relation
  - gcd, lcm
  - $a = b \pmod{n}$

Week 2: Sets, Relations and Functions

Sets, languages, relations and functions:

- Set notation: $\emptyset$, $\subseteq$, $\{\ldots\}$, $[\ldots]$
- Set operations: $\cap$, $\cup$, ^{c}, $\setminus$, $\oplus$, $\times$
- Set theory laws

- Symbols, words, languages
- Language definitions: Σ^*, $\mathrm{length}()$, concatenation

- Relational image ($R(A)$), converse relation ($R^{\leftarrow}$), inverse image ($R^{\leftarrow}(B)$), relation composition ($R;S$)

- Domain, co-domain, image, function composition
- Surjective, injective and bijective functions

Week 2: Need to know

- Set operations
- Proofs using laws of set operations
- Definitions of languages, relations and functions
- Relational/functional image, relational/functional composition

Week 3: Relations and Functions

- Binary relations:
  - Equivalence relations, equivalence classes
  - Partial orders, Hasse diagram, lub, glb
  - Total orders, topological sorting, lexicographic, lenlex
- Functions:
  - Injective, surjective, bijective
  - Inverse function
  - Matrices

Week 3: Need to know

- Binary relations:
  - Equivalence relations, equivalence classes
  - Partial orders, Hasse diagram, lub, glb
  - Total orders, topological sorting, lexicographic, lenlex
- Functions:
  - Injective, surjective, bijective
  - Inverse function
  - Basic Matrix operations: $A + B$, $A \cdot B$, A^T

Weeks 4 & 5: Big-O, Recursion and Algorithmic Analysis

- Recursive definitions
- Solving recurrence equations
  - Unwinding
  - Simplifying using big-O
  - The Master Theorem
- Algorithmic analysis: Big-O, Ω and Θ
  - Worst-case analysis
  - Running time (number of elementary operations)
  - Using big-O to simplify analysis
  - Aim to compare with "simple" functions, e.g. $O(n)$, $O(n \log n)$, $O(2^n)$, etc

Weeks 4& 5: Need to know

- Big-O vs Big-Ω vs Big-Θ
- How to define things recursively
- (At least) One method for solving recurrences
- Algorithmic analysis

Weeks 5 & 6: Induction

- Basic Induction
- Induction variations
- Structural Induction

Weeks 5 & 6: Need to know

How to do proofs using:

- Basic induction, from m, steps ≥ 1.
- Structural induction

Week 6: Propositional Logic I

- Propositions and connectives
- Syntax: Well-formed formulas
- Semantics: Boolean algebra $\mathbb{B}$
- Logical equivalence
- Entailment and validity

Week 6: Need to know

- Formal definitions of Propositional logic
- Evaluating functions on well-formed formulas

Week 7: Propositional Logic II

- Boolean functions
- CNF and DNF
  - Converting formulas to CNF/DNF
  - Canonical DNF/CNF
  - Karnaugh maps
- Boolean algebras
- Beyond Propositional Logic

Week 7: Need to know

- Definitions: Boolean function, CNF, DNF
- One method of creating an equivalent CNF and DNF
- Boolean Algebras:
  - Definitions
  - Giving proofs

Week 8: Graphs

- Definitions and notation: vertices, edges, paths, cycles, connectedness
- Isomorphisms, Automorphisms
- Important graphs: Trees, Complete graphs, complete k-partite graphs
- Graph traversals: Eulerian path/circuit, Hamiltonian path/cycle
- Graph properties: Chromatic number, Clique number, Planarity

Week 8: Need to know

- Definitions and notation: vertices, edges, paths, cycles, connectedness
- Important graphs: Trees, Complete graphs, complete k-partite graphs
- Graph traversals: Eulerian path/circuit, Hamiltonian path/cycle
- Graph properties: Chromatic number, Clique number, Planarity

Weeks 8 & 9: Combinatorics

- Disjoint sets, inclusion-exclusion rule
- Cartesian products, counting sequences
- Permutations and Combinations
- Sequences with/without replacement and with/without ordering, Balls in boxes

Weeks 8 & 9: Need to know

- Counting based on basic set operations (union, intersection, etc)
- Four different types of sequences, their notation, and how to count them (i.e. formulas)

Weeks 9 & 10: Probability

- Sample space
- Probability distribution, uniform probability
- Combining Events
  - Events from the same set of outcomes
  - Events involving different sets of outcomes
- Independent events
- Recursively defined scenarios
- Conditional probability
- Expectation
- Variance

Weeks 9 & 10: Need to know

- Definitions and notation
- Probability computations:
  - Basic: First principles, examining sample space
  - Complex: Combining events: union, sequencing, conditional, recursive
- Expectation computations