

Exam Information
&
Unit Conclusion
(Week 13 Lecture)



Lecture Outline

- Basic exam information
- What will and will not come out in the exam
- Concluding remarks about this unit



Exam Venue, Time and Date

- Time & Date
 - 9:30am 24th June 2004
- Venue
 - ECL Lecture Theater 2.
 - Rockingham ACL 3.17.
- Duration
 - 3 hours + 10 minutes reading time



Exams Page

- URL

- <http://www.it.murdoch.edu.au/units/ICT336/exam/index.shtml>

- All the information on that page I will cover today in this lecture.



Past Exam Questions

- Many past year exam questions from 2001, 2002 and 2003 are relevant.
 - Anything relating web server/client, Apache, XML and Web Services.
- The questions in this year's exam **will not be the same**, but the style and technical difficulty will be similar.
- Ignore questions on
 - PHP, ASP and server-side web site programming.
 - Wireless programming (eg. the long question on WAP in the 2001 exam is not relevant).



Marks

- Total marks of the exam paper: 169 marks.
 - Part I:
 - 14 questions
 - Short and medium length answers
 - questions worth anything from 1 to 10 marks
 - Part II:
 - 2 questions
 - Long answers
 - 20 marks each.



Answering Questions

- Write the answers on the question paper itself.
- There are ample space provided.
 - Do **not** use the space provided as an indication of the length of the answers required. Read each question carefully and answer them directly.



Distributions of Marks

- By topics:
 - Programming in Perl (not including topics below): 7%
 - Web Server administration with Apache: 13%
 - Web servers and web client programming: 29%
 - XML: 40%
 - Web Services: 11%
- By type:
 - Conceptual questions: 59%
 - Practical questions: 41%



What will DEFINITELY come out in the exam

- The following Practical Questions are in the exam:
 - Picking mistakes from XML document.
 - Picking mistakes from DTD or Schema.
 - Give outputs for, explaining and/or adding to:
 - Perl code for web server and web client given in the lectures.
 - XSLT style sheets.
 - Perl code for XML processing using XML::DOM.
 - WSDL or SOAP::Lite code.



What will DEFINITELY come out in the exam

- The list in the previous page **is not comprehensive!**
 - There **may be other types** of practical questions which will appear in the exam which are not on the list.
 - It is to let you know what you must **at least** study for.



What will DEFINITELY NOT come out in the exam

- You will **not** be asked to write **large chunks of original code from scratch.**
 - You may, however, be asked to add a few lines to existing code to do certain things, but questions like those you can always look at the existing code to derive the new code.
- You will also **not** be asked about details of example XML applications
 - but you may choose to use them to support your answers about concepts of XML



What will DEFINITELY come out in the exam

- Conceptual Questions in the exam will be one of the following types:
 - Explain the purpose of a technology (ie. what benefits does it give us).
 - Describe how a technology works.
 - Explain the similarities and differences between one technology and another.
 - Explain how one technology is related to another (specifically how they depend on each other).
 - How to apply a technology to do a certain thing.
 - Acronyms (see first question of past year exams).



What will DEFINITELY come out in the exam

- There are **no other types** of conceptual questions beside the ones listed in the previous page.



Lectures this semester

- All lectures except the first and last (this lecture) are examinable.



Labs this semester

- All labs are relevant and important for the programming questions.



Assignments this semester

- The exam questions will not be as tough as the questions in the assignments.
- If you can do the assignments, then you will do very well in the programming parts of the exam.



Learning Objectives for this unit (in a nutshell)

From our first lecture:

1. Understand technologies.
2. Write software.
3. Construct solutions.
4. Don't depend on spoon-feeding.



Programming Internet Systems

- The basic process of creating systems for the Internet would follow the first 3 objectives.
- Understanding the basis on the process will stand you in good stead on the future
 - Where to find information, who to ask
 - What overall concepts to learn first, what details to learn later as required
 - How to glean key things from examples
 - When you actually need to read specifications
 - etc



Programming Internet Systems

- After you graduate, you will no longer have a unit structure and material to guide you in that process.
 - Learn to identify those components yourself



The Ability to Manage Change

- Finally, as my concluding point for this unit, I can't emphasize the unit's Learning Objective 4 enough.
- Your most (and perhaps only) dependable skill you can take for the rest of your career in Internet development is **the skill to teach yourself**.
 - I don't know what the long-term future is going to be like, I only know it is going to be very different from what it is now.
 - The people who thrives are the ones who are not left behind.



Never assume you know

- A true anecdote: Most of our IT students who takes 1st year units in their 3rd year usually averages a C for those units.
 - They start the semester knowing 50%-60% of the material, therefore think it is easy, and so end the semester still knowing only 50%-60% of the material.
 - While good “naïve” 1st year students start the semester with 0%, and end up knowing 80%-90% of the material.
- In this field, never say “I know” - only say “What else can I learn”