CM0340 Multimedia Revision Guide

Detailed below is a list of aspects of the course that are *not* going to be examined and parts of the course that are likely to be examined.

Back

Close

The Best Revision Aid

Past exam papers and indicative solutions for some most relevant exam papers are available on the Blackboard.

The best way to practice for the exam is to use this to test yourself and also for the style of exam questions and what the *indicative* solutions expect.

Please note the amount of marks available per question part — as an indicator to the amount of attention you should devote to your solution.




The Best Revision Aid (Cont.)

This year's exam paper follows the same patten as almost all past papers:

- 4 Questions in total, each with 27 marks.
- You answer **3** out the 4.
- Each question in three broad parts
  1. Basic bookwork definitions
  2. Bookwork description of theory or algorithm
  3. Unseen extended reasoning part extension of first two parts. Often enumerated example based on above.

  – occasionally the above parts broken into smaller parts.

What you DONT have to learn — NOT IN EXAM

- Basic historical facts as mentioned in the lecture notes. *E.g.* Dates of algorithm development, MPEG video history,
  Dates of Sythesisers/Synthesis methods
  - **Note**: You are expected to note difference between different algorithms that may have developed from previous version. *E.g.* Differences between LZW and LZ, MPEG-1 (and 2) video and H.261.
  - Other related historical "trivia" related to certain algorithms, synthesis methods are not required. *E.g.* When/where sythesis methods first used in hardware/software implementations, Development of MPEG video and audio, Dolby AC.




Back
Close

What you DONT have to learn — NOT IN EXAM

- MATLAB. You do not necessarily need to know basic MATLAB syntax for the exam.
  - There will be **NO** specific exam question on MATLAB programming, graphics or GUI design in the exam.
  - **Note**: Exam questions will ask for a suitable algorithmic description. In many cases giving MATLAB code (or fragments) might be the best solution and maps most closely to what has been described in the lecture notes.
    IN SUCH CASES giving MATLAB code as part of your solution is perfectly valid.
    ALTERNATIVELY you may use pseudo code to describe the algorithm as long as it is understandable.




What you DONT have to learn — NOT IN EXAM (Cont)

- MATLAB Cont.:
  - I would also advise looking at the MATLAB code examples, running them and trying out variations as suggested in Lab classes as a way of understanding more deeply almost all aspects of the course. Also the latter lectures explaining BASIC MATLAB implementations of algorithms of discussed throughout the course to hopefully gain a better understanding of them.
  - You can find some hints for the lab exercises in *Blackboard*$\rightarrow$ *Module Documents* $\rightarrow$ *Lab Classes* which may help you get a better understanding.




What you DONT have to learn — NOT IN EXAM (Cont)

- Complicated computational (or mathematical) enumerations of algorithms/definitions. YOU WILL NOT BE EXPECTED to evaluate a Fourier or Discrete Cosine Transform of a 1D or 2D signal, as demonstrated in lecture notes and handouts — this would simply take too long in an exam.

NOTE: Other questions in the exam could test for understanding and definitions of Fourier or Discrete Cosine Transform though.




What you DONT have to learn — NOT IN EXAM (Cont)

- Basic Calculus based derivations of some examples discussed in lecture notes. *E.g.* FFT of Sinc function, Convolution integral example will NOT be examined.
- Detailed facts relating to certain areas of the notes will not be required to be memorised,
  - *E.g* exact MIDI commands, General MIDI instruments numbers SOME IDEA of how MIDI works is required and has been examined in the past.
  - *E.g* Exact numbers on formats such PAL/NTSC resolution, number of bits used in various MPEG compression,
  - *E.g* JPEG quantisation tables

What you DONT have to learn — NOT IN EXAM (Cont)

- The following topics largely covered in CM0268 will NOT be examined *by themselves*:
  - Signal diagrams
  - Z-transforms
  - Filter design (Finite Impulse Response/Infinite Impulse Response)
  - Digital Audio Effects
- However, you should have a sufficient understanding of these when they are applied to e.g. digital audio synthesis.




Back
Close

What you DO have to learn — DEFINITELY VALID FOR AN EXAM QUESTION

- All basic definitions, including mathematical definitions.
  E.g Fourier or Discrete Cosine Transform
- Basic descriptions of algorithms for all described in lectures. This may be via pseudo-code and/or diagrams and/or MATLAB code fragments.
- Broad differences between similar algorithms. *E.g.* JPEG/MPEG I-frames, H.261/MPEG video, MPEG Audio/Dolby AC, etc.
- How to apply algorithms to enumerate example data streams — **Especially** independent compression algorithms — Huffman coding, arithmetic coding, LZW etc.
  Exception Fourier/Discrete Cosine Transforms, Complete JPEG/MPEG encoding
- See exam papers for examples of questions.




Back
Close

Calculator

- Please bring a calculator to the exam as there might exist some calculations that you can benefit from having a calculator.
- University regulations of 'Use of Calculators'
- The use of scientific calculators is permitted in all examinations.
- The calculators which students may use in examinations must be noiseless, battery or solar-powered, scientific calculators with numeric displays only. Programmable calculators, or calculators with an alphabetic keyboard and/or the ability to store and retrieve text are not permitted in any examination.




Back
Close

Revision Lecture

- One revision lecture at 13.10pm Monday 9th January (12th Week).
- Email discussion: dave.marshall@cs.cardiff.ac.uk
- **Discussion Board** on Blackboard
- Submitted coursework will be returned in Revision Lecture.




Back
Close