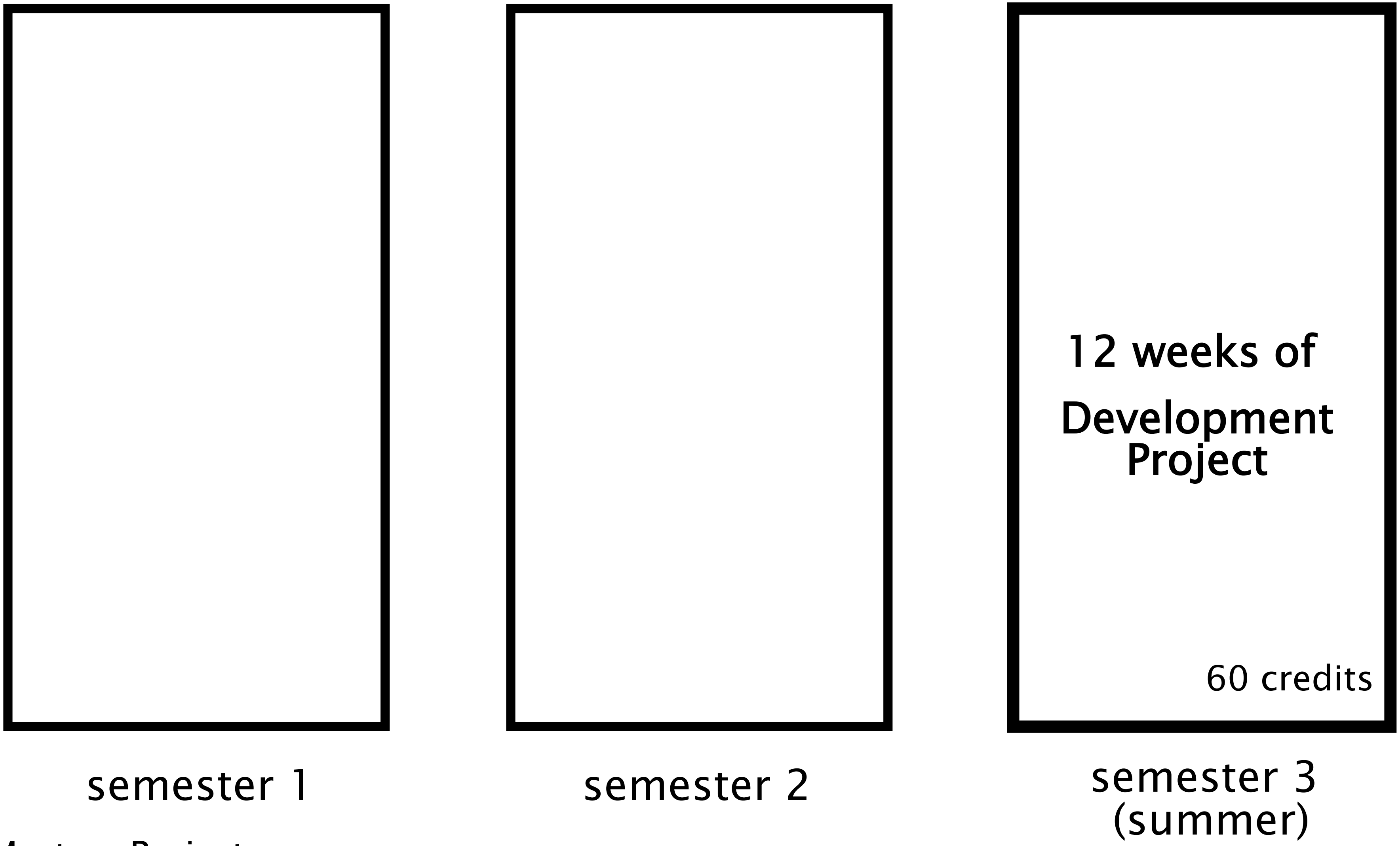
Masters Projects

Conducting your masters project

Prof Joemon Jose

School of Computing Science
University of Glasgow

MSc development projects overview



12 weeks of

Development Project

60 credits

semester 1

semester 2

semester 3 (summer)

Impact on your results

Your project is critically important...

Master degree

- taught stage ≥ 12.0 and project at least **D3**

Master degree with merit

- taught stage ≥ 15.0 (normally) and project at least **B3**

Masters degree with distinction

- taught stage ≥ 18.0 (normally) and project at least **A5**

To clarify for progression to the project: a total of at least 120/130 credits (130 for MSc-IT and MSc-SD)

- at least 90 credits at level M
- at least 90 credits are grade D or above
- no credits at grades G or H
- GPA ≥ 12.0

- If not sure check with Helen/Simon

Once you submitted your thesis

- Examiners
  - Supervisor (involved in the project work)
  - Reader (independent of the project work)
- Both mark reports & code independently
- If their marks differ by 1 band
  - Supervisor mark is the final mark
- If their mark differ by more than 2 bands
  - They discuss and agree on a mark
- If they cant agree
  - A third marker will be appointed
  - All three discuss and agree

Submission

Your dissertation has a strict page limit of 30 including Appendix

- Recommended page length - 20
- two band penalty for non-adherence to submission instructions
- previous A grade dissertations on the moodle for reference
  - Check out Hall of Fame section
  - some these are 40 pages, so watch out!

Plan to start writing report at least 3 weeks before the deadline

The deadline is Friday 06 September 2019 at 16:00

- late submission will be penalised
- Getting extensions is a difficult process – it will affect the results

If you have any problems please discuss with your supervisor first and if not resolved contact me as soon as possible

Project coordinator

- Make sure process is run smoothly
- Allocate students to supervisors
- Make sure marking is done according to the procedures
- Finalise the marks and forward to the exam board
- I don’t have much handle on
  - When Exam board is organised
  - When the results are announced (Teaching Office)
- Once you submit your report
  - And final marks completed
  - My responsibility ends
- Course Directors
  - Custodians of the course
  - Anything deviating from agreed procedures need their explicit permission

Important dates for MSc Development Projects (2018-2019)

- **MSc Development project (start - finish):**
  - 17 June 2019 - 06 September 2019
- **MSc development project (Part-timers 2018-2019 (2nd year part-time students)) deadline is :**
  - Friday 06 December 2019
- **MSc development project (Resit students 2018-2019)**
  - After the exam you need to contact Course Director before starting the project
  - 9 September 2019 - Friday 06 December 2019
- **Levels of support**
  - 15-30 minutes supervisory time
  - Lab tutor (Douglas Fraser for IT/SD/CSE) 12-5 Monday , Wednesday, Friday
  - Supervisors could be away part of time
    - Discuss in advance; plan your time

What you submit

- Report (30 pages)
- Code (archived/compressed)
- Other materials like documentation, if you have developed
- Details are on Moodle
- Please note you should be physically in the computing science laboratory to submit
- If you are away by this time
  - Place all the above material in google/dropbox folder and mail your supervisor and get their help to upload
  - Supervisor will forward your mail to support to upload
  - Ask for a confirmation

Extensions/Jobs/..

- **Anything out with stipulated period/programme**
  - Need Course Directors explicit Approval
- **Extensions**
  - Getting extensions is a difficult process –
  - it will also affect the results are released!
  - Discuss with your supervisor first
  - After the agreement, mail supervisor an email along with details
  - Supervisor to forward the email along with the recommendation to course director or me
  - Any other requests will not be answered
- **Jobs**
  - Need respective course director's approval
- **Communicate**
  - Always state your course

Suggested Timeline

- **June 17, 2019 – Start the project**
  - Meet up with supervisor
  - Agree on a topic
- **June 30, 2019**
  - Specify the project; update the title
- **July 15, 2019**
  - Draft of Analysis Chapter
- **July 30, 2019**
  - Draft of Product Chapter initiated
- **August 15, 2019**
  - Evaluation started
- **August 30, 2019**
  - Development stop; Concentrate on writing

REPORT/EVALUATION

Evaluation Criteria

Analysis 15%

- surveyed relevant literature and existing software products, captured the requirements, analysed the problem and devised a suitable approach

Product 40%

- product well-designed, functional, reliable, robust, efficient, usable, maintainable, and well-documented and demonstrated

Evaluation 15%

- software tested and user evaluation, suggestions for further work

Report/Dissertation 20%

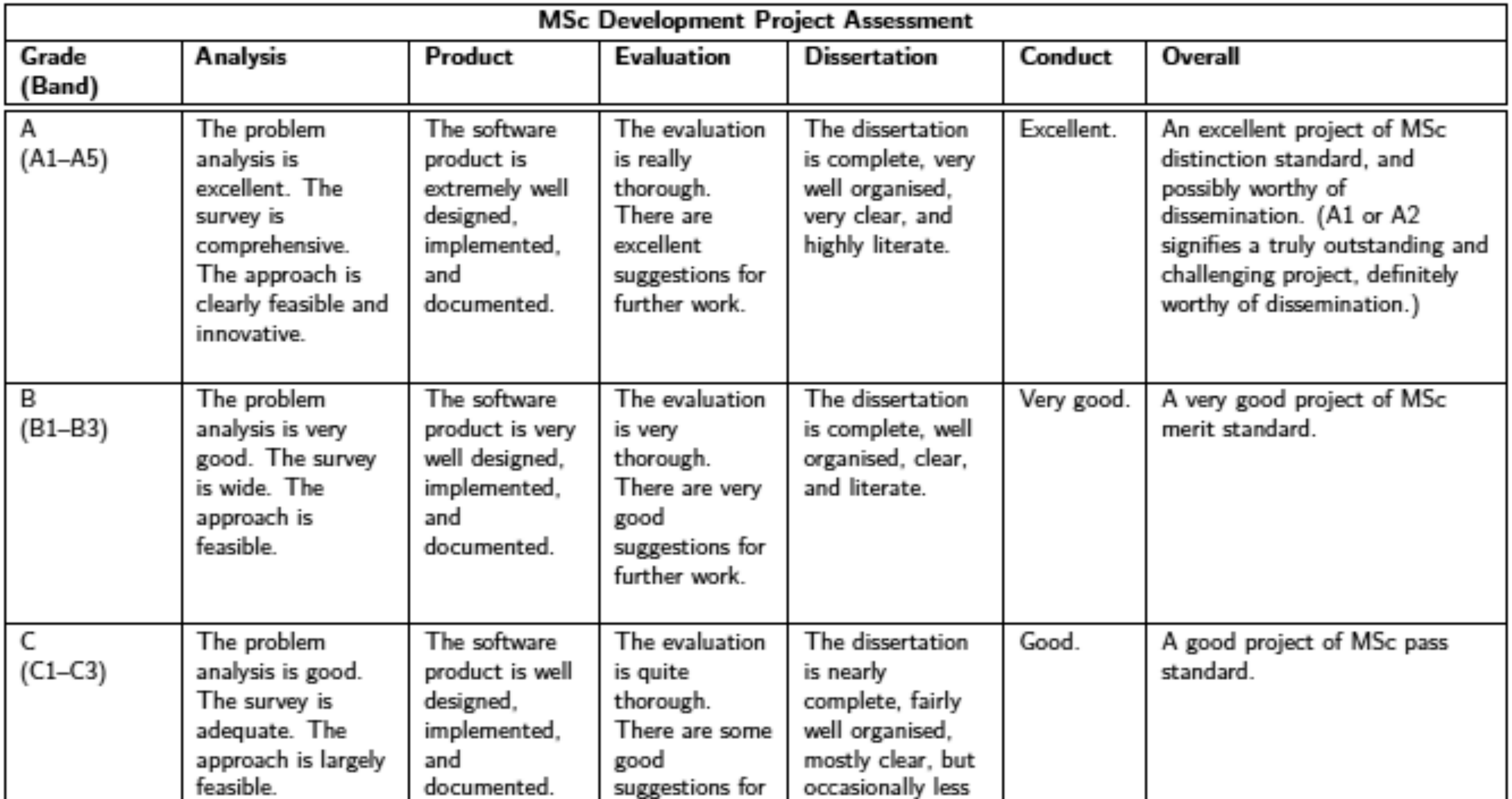
- complete, well-organised, clear, and literate, clearly explain the steps of the project with bibliography and proper citations

Conduct 10%

- did the student attend meetings & engage effectively with the supervisor?

MSC development Project assessment criteria

- http://moodle2.gla.ac.uk/pluginfile.php/485718/mod_resource/content/5/msc-assessment.pdf



MSc Development Project Assessment						
Grade (Band)	**Analysis**	**Product**	**Evaluation**	**Dissertation**	**Conduct**	**Overall**
A (A1–A5)	The problem analysis is excellent. The survey is comprehensive. The approach is clearly feasible and innovative.	The software product is extremely well designed, implemented, and documented.	The evaluation is really thorough. There are excellent suggestions for further work.	The dissertation is complete, very well organised, very clear, and highly literate.	Excellent.	An excellent project of MSc distinction standard, and possibly worthy of dissemination. (A1 or A2 signifies a truly outstanding and challenging project, definitely worthy of dissemination.)
B (B1–B3)	The problem analysis is very good. The survey is wide. The approach is feasible.	The software product is very well designed, implemented, and documented.	The evaluation is very thorough. There are very good suggestions for further work.	The dissertation is complete, well organised, clear, and literate.	Very good.	A very good project of MSc merit standard.
C (C1–C3)	The problem analysis is good. The survey is adequate. The approach is largely feasible.	The software product is well designed, implemented, and documented.	The evaluation is quite thorough. There are some good suggestions for	The dissertation is nearly complete, fairly well organised, mostly clear, but occasionally less	Good.	A good project of MSc pass standard.

Structure of the Report

First decide the structure of your dissertation:

how it will be divided into chapters (and appendices)

You should subdivide chapters into sections

You might even subdivide some sections into sub-sections

- Report should represent the *final* version of the project
- Title page:
- including (in order from top to bottom), your project title, your name in full; and the month and year of submission.
- Chapter 1 Introduction – ~1 page
- briefly explain the context of the project problem.
- Specify overall aim and objectives and report structure

Requirements chapter:

- Statement of Problem : clearly state the problem to be addressed in your project. Explain/Motivate why it would be worthwhile to solve this problem.
  - The problem statement is excellent and fully justified.

- Background Survey/Analysis: present an overview of relevant previous work including articles, books, existing software products and requirements identification. Critically evaluating their strengths and weaknesses.
  - The survey is comprehensive and incisively critical

- Effectively combine above in one chapter- articulation is something like: I have a problem (statement of the problem) and here is the background
- Or here is the background and I am going to solve the following problem

“A1-A5” :: The problem analysis is excellent. The survey is comprehensive. The requirements are clearly specified. (15% weightage!!)

Thesis /report Structure -1

Chapter 2 Analysis/Requirements ~4-6 pages

Discuss the background (1 page)
Explain how the requirements are gathered (the tools and techniques used – ½ page to 1 page)
Analysis ~1-2 pages
Specify the aims and Objectives - Ordered list of Features and justification (~2 pages)
User stories and MOSCOW statements in the Appendix

(Marking) A1-A5 :: The problem analysis is **excellent.** The survey is comprehensive. The approach is clearly feasible.

Content – Development project

In your design & implementation chapter:

- discuss the main features of your design and how it evolved; include high-level design diagrams/data-flow diagrams
- highlight any novel features; explain critical design decisions
- but don't include design documentation here (this can be a separate document)

In your implementation part:

- discuss the main algorithms and data structures and how they evolved
- highlight any novel features
- also discuss your testing strategy

Product – 40% weightage

product well-designed, functional, reliable, robust, efficient, usable, maintainable, and well-documented and demonstrated

Thesis /report Structure -2

Chapter 3- Design & Implementation ~4 – 5 pages

System architecture major design decisions and rationale (~2-3 pages)
Implementation details ~1 page
Screenshots or other similar things ~1 page

Additional details; design diagrams in the Appendix

(Marking) A1-A5 :: The software product is extremely well designed, implemented, and documented.

Content – Development project

In your evaluation chapter:

- describe how you evaluated your product
- summarise the evaluation results, and use them to critically evaluate your own work
- be honest about any shortcomings

Evaluation – 15% weightage!!

In your conclusion:

- describe the status of your product
- summarize what you have achieved
- compare to what you originally set out to achieve
- relate your work to relevant previous work
- suggest further work that you think would be worthwhile

Thesis /report Structure -3

Chapter 4- Testing & Evaluation details - ~4-5 pages

Software Testing – strategy & statistics 1-2 pages
Explanation of Evaluation Strategy (1/2 page).
Evaluation Results – 1-2 pages
(4/5 pages covering each aspect of evaluation and discussion of results.)

(Marking) A1-A5 :: The evaluation is really thorough. There are excellent suggestions for further work.

Thesis /report Structure -3

Chapter 5 – Conclusion – 2-3 pages

Conclusions should state the achievements, and a reflection on achievements (what wasn't achieved/ why and what more could have been done or done differently in hindsight - pick one or two issues to discuss in *depth* rather than trying to be comprehensive.

should include future work as well.

- *Recommended Pages – 20 Pages*
- References/Bibliography – ~1-2 page

list, in alphabetical order by author and date, all articles that you have consulted.

- Appendix (any number of pages)
- Absolute Maximum including Appendix ; 30 pages

Marking Procedure – Grade descriptors

Supervisor and a READER marks the project	A–B–… C/D
A **The dissertation is complete, very well organised, very clear, and highly literate.** **20% weightage**	Excellent: MSc distinction standard. (A1 or A2 signifies exceptionally good work, definitely worthy of wider dissemination.)

Chapter 6. Bibliography:

  - list, in alphabetical order by author and *date, all articles that you have consulted. For each article, give full bibliographic details*

- **Use consistent style**
  - Full bibliographic details for all items
  - Web pages – access date
  - Books/articles – full details; use APA style- readability is high
- **Collect all the details when you access a document first**
  - Update the .bib file or Endnote
    - Provide a summary in the dissertation, so that you can reuse it later

<u>The report is complete, very well organised, very clear, and highly literate.</u>

Content – Abstract/Summary

The abstract is a short summary of the dissertation

Its purpose is to catch the reader's attention: is this dissertation worth reading in full?

It should be ½–1 page long

It should briefly outline the context of the project, its goals, and its achievements

It should highlight any novel aspects of the project

Content – Table of contents

The table of contents lists the chapters of the dissertation

- showing each chapter’s number and title, and the number of its first page

Similarly, it lists the abstract, acknowledgements, appendices, bibliography, …

If chapters are subdivided into sections, these should also be listed

- showing each section’s number and title and the number of its first page
- section and sub-section details should be indented and less prominent

- If you use a template, this should come automatically

RECOMMENDATION

Steps

- **Create a project template**
  - You can get templates on moodle
    - Use **Latex** or Word
  - This help your formatting easy and simple
- **Use a bibliography**
  - Endnote
  - .BIB FILE
- **Use picture drawing packages consistently**
- **Create place holders for each chapter**
  - Compile your template
- **Already got an empty project report**

<u>The report is complete, very well organised,
very clear, and highly literate.</u>

Content – Avoiding plagiarism

Plagiarism means:

using another person's work without acknowledgement

- i.e. presenting another person's work as if it were your own

You must cite the source of anything that is not your work, including:

- text (either direct quotation or paraphrase)
- ideas
- designs
- code
- data
- diagrams, images, etc

Content – Avoiding plagiarism

If you must use another person’s words exactly, include quotation marks as well as a citation

Almost always it is better to paraphrase the other person’s words (using your own words) – but still include a citation

“Testing can prove the presence of errors, but never their absence.” (Dijkstra 1968)

direct quotation

Dijkstra (1968) observed that testing might expose errors in a program, but no amount of testing can ever prove the program free of errors.

paraphrase

Content – Plagiarism vs acknowledgement

Every project builds on previous work

It is normal to use previous work in your project, but you are also expected to contribute something new

- you will be assessed on your own contribution

Whenever you use another person’s work, you must acknowledge its source

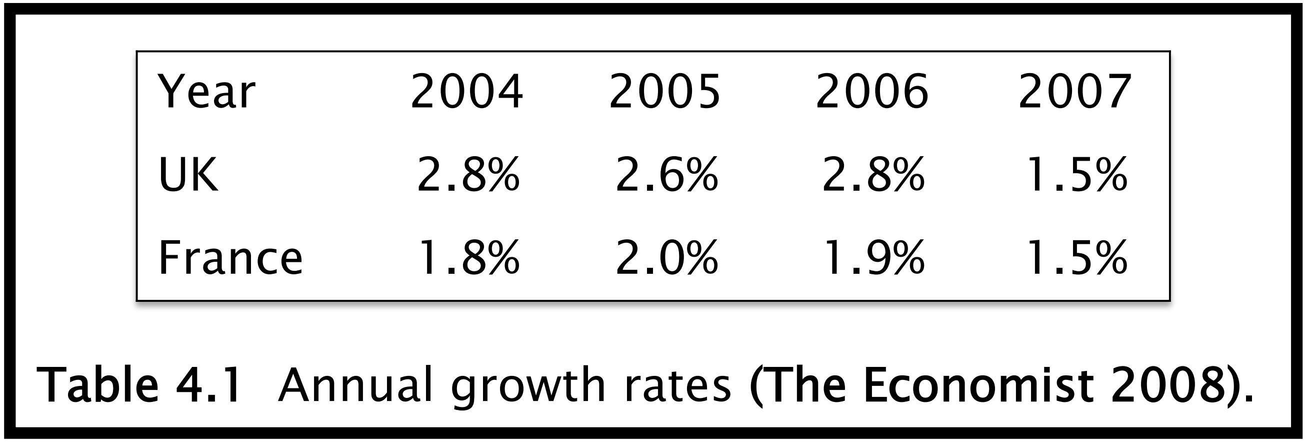
Failure to acknowledge a source is plagiarism

- this means presenting another person’s work as if it were your own

Content – Plagiarism vs acknowledgement

Wherever you reuse another person’s code, acknowledge the source in the code itself (as a comment) and in your dissertation

Wherever you use another person’s idea, design, data, table, figure, image, …, acknowledge the source in your dissertation. E.g.:



Year	2004	2005	2006	2007
UK	2.8%	2.6%	2.8%	1.5%
France	1.8%	2.0%	1.9%	1.5%

Table 4.1 Annual growth rates **(The Economist 2008).**

Content – Bibliography

The bibliography must list all sources (books, articles, web sites, etc.)

- that are cited in your report
- that provide useful background information

Normally order the sources by authors' surnames and dates

For every source, include the author names, date, title and

- for an **article**: the title of the journal or conference record in which the article was published, and page numbers
- for a **book**: the name of the publisher
- for a **web site**: the URL
- also, if useful, briefly summarize the content

Content – Citations

The Harvard style: author_surname, date

The quick-sort algorithm was invented by Hoare (1962); see also Wikipedia (2007).

Python (Downey et al. 2002, Martelli, 2006) is a highly dynamic language, suitable for object-oriented and functional programming.

citations in the text

Downey, A., Elkner, J., Meyers, C. (2002) *How to Think Like a Computer Scientist – Learning with Python*, Green Tea Press.

Hoare, C.A.R. (1962) Quicksort, *Computer Journal 5*, pp. 10–15.

Martelli, A. (2006) *Python in a Nutshell*, O' Reilly.

Wikipedia (2007) Quicksort, en.wikipedia.org/wiki/Quicksort.

bibliography

Content – Citations

The Vancouver style (numbered) is an alternative

The quick-sort algorithm was invented by Hoare [2].

Python [1, 3] is a highly dynamic language, suitable for object-oriented and functional programming.

citations in the text

[1] Downey, A., Elkner, J., Meyers, C. (2002) *How to Think Like a Computer Scientist – Learning with Python*, Green Tea Press.

[2] Hoare, C.A.R. (1962) Quicksort, *Computer Journal 5*, pp. 10–15.

[3] Martelli, A. (2006) *Python in a Nutshell*, O'Reilly.

bibliography

Latex supports automatic styles (bibtex), while MS Word does not

Content – Supplementary material

Supplementary material includes code, documentation, detailed evaluation results, …

Do not include supplementary material in chapters of the dissertation

Put supplementary material:

- in an appendix but **only if**
  - it is essential to understanding of the dissertation
  - **and** it is not too bulky
- if it is not an appendix include separately in the electronic submission

Include references to the supplementary material where necessary in your dissertation

Ethical approval

If your project involves the participation of other people (for example in an evaluation), or data relating to other people, then you should complete an ethics checklist form

You may also need to apply for approval from the schools ethics committee see: http://www.dcs.gla.ac.uk/ethics for further details