

The Computer Science and Informatics Fourth Year Project: Rules and Advice

Mel Ó Cinnéide (edited by Eleni Mangina)

August 22, 2011

1 Introduction

The fourth year project constitutes a vital component of your degree and as such requires careful consideration and effort. Every final year honours and joint honours student from both Science and Arts is obliged to undertake a final year project. The purpose of the project is to introduce the student to a particular field of Computer Science and to give them an opportunity to learn how to undertake a major project, taking it from problem specification through to problem solution.

The purpose of this document is to outline the nature of a fourth year project, to explain the procedures involved in choosing and undertaking such a project, and to provide guidelines on the important task of report writing. Please bear in mind that this document is intended only to describe the process and the marking scheme in an indicative way, and in appropriate circumstances certain details may be changed during the year.

2 Communication and People

We may communicate with you in various ways regarding project guidelines, deadlines etc. Announcements may be made in lectures, emailed to you or posted in the fourth year projects web page (<http://www.csi.ucd.ie/content/fourth-year-project-information>). Be sure to keep yourself informed of what is going on.

2.1 The Project Review Committee

The Project Review Committee comprises a small number of members of academic staff, none of whom are involved in project supervision.

The responsibilities of the Project Review Committee include vetting all project specifications, deciding if a project should be pre-allocated to a student, marking student presentations and deciding the final report mark. This committee may also handle other project-related matters that arise during the year. If you wish to make a submission to this committee on any matter, please do so via the Project Coordinator. Do not contact members of this committee directly.

2.2 The Project Supervisor

Your project supervisor plays a vital role in guiding and advising you as you work on your project.

You should be sure to make good use of your supervisor, and to let them see that your approach to your project work is organised and competent. Typically you should meet your supervisor about once a week for about 30 minutes. Be professional: if you cannot make a meeting be sure to inform them in plenty of time. Don't come to your supervisor with every little problem you have - try to work things out yourself so you can have an informed discussion with them about the problem. At the same time, if you are having serious problems with your project and progress is halted, be sure to let your supervisor know as soon as possible. Sometimes project goals have to be redefined, and the earlier this happens the better.

If you feel that the supervision you are receiving is not sufficient or appropriate, the best person to speak with in the first instance is your supervisor. If you feel this is not possible, or you do it and find it not fruitful, you should discuss the matter with the Project Coordinator, the 4th Year Coordinator, or the Head of School.

You will hopefully come to regard your supervisor as a friendly and supportive figure. If you have any life problems during the year that are impinging on your work, let them know. In circumstances such as illness or bereavement, we are flexible in the allowances we can make for final year students.

2.3 The Project Coordinator

The project coordinator is responsible for all aspects of the organisation of the final year projects. The current project coordinator is:

Eleni Mangina
Room B2.05
email: eleni.mangina@ucd.ie

The role of the coordinator is to ensure all aspects of the project process are adhered to. The coordinator, together with the Project Review Committee ensures that standards are maintained. The coordinator is also a useful source of advice relating to projects generally and is available for consultation on any project matters that are of concern to you.

3 Choosing a Project

3.1 Normal Allocation of Projects

On the fourth year project web page you will find a link to a page that contains descriptions of the projects that

are available this year. There is a broad range of projects available and you should find many there that are of interest to you.

On the project web page you will be invited to select your top N projects (usually 10). It is vital that you give this matter some consideration. If you are the only person who has chosen a certain project, you are likely to be allocated this project. In the more typical case where several students vie for the same project, a random allocation takes place. If for example you only choose three projects and these all turn out to be popular projects, you might not be allocated any of them. It is therefore important to make a careful selection of all N projects.

The allocation only takes into account the list of projects you select. Your results in third year, whether you are in Science or Arts and exactly when you made your selection (assuming you were on time) all have no bearing on the process.

Another aspect of project selection is of course the project supervisor (see section 2.2). Many of the lecturers you already know from your years as an undergraduate. Other lecturers you may not have encountered before, and in this case you may wish to discuss their projects with them before making a selection. Obviously the type of supervisor you work with best is a very individual matter.

The School aims to allocate projects as early as possible. The deadline for project selection will typically be the first week of the first semester.

3.2 Student-driven Project Specification

Some project specifications may be (at least partially) defined by students who will be coming into 4th Year. If interested, a student should contact a suitable staff member as soon as possible after the end of 3rd Year to discuss this. If the staff member is willing, they will then propose the project and request that it be assigned to the student. The Project Review Committee (see section 2.1) makes these decisions, not the proposed supervisor.

Having a hand in defining your own project is very positive in certain ways, but it requires a lot of effort on your part initially, and there is no guarantee that the final project proposal will be accepted by the Project Review Committee.

3.3 Pre-Allocation of Projects

If you see a project that you are *particularly* interested in, and exceptionally suited to, you may make a case for this being assigned to you by emailing the 4th Year Project Co-ordinator. Having the support of the project's supervisor would strengthen your case. Such applications will be accepted only in exceptional circumstances, and the decision is made by the Project Review Committee (see section 2.1), not by the project supervisor.

3.4 If you're unhappy with your allocated project...

Once the allocation of students to projects has been made, students have 2 days to make a case for modifying their

project and/or moving to another (willing) supervisor. These decisions are made by the Project Review Committee (see section 2.1), not the staff members involved. Such applications will be accepted only in exceptional circumstances.

4 Allocation of Marks

Your project mark comprises 30% of your final mark in Computer Science. The breakdown is typically as follows:

Project Interim Report (includes Project Management Deliverable):	10%
Project Presentation:	15%
Project Report and Outcome:	65%
Conduct of Project:	10%

The marks for the Interim Report and the Conduct of Project are awarded by the supervisor. The Project Presentation is marked by the PRC. The Project Report and Outcome is assessed by the supervisor, and the detailed comments are given to the PRC. The PRC take these comments into account when grading the project, and also give comments with the assigned grade to the project co-ordinator.

4.1 Project Reports

There are two reports which you have to write as part of your project work. The *Interim Report* is due about half way through the project, and the *Final Report* is due at the end of the project. The *Project Management deliverable* is included in, and is part of, the Interim Report.

All reports should be produced in a professional manner. Section 10 provides detailed guidelines and advice on the writing of the final report, much of which applies to the other reports as well. You will be provided with a Word document and LaTeX template that define the formatting and style of your reports. You can choose which of these to use, or indeed any other word processing package. This saves you a lot of work, and introduces a standard layout to all the reports. These template documents are to be used in writing all your final reports.

4.2 Critical Dates

For the Academic Year 2011-2012, the critical dates that you should be aware of are:

Project Selection:	Wed Sept 14
Project Allocation:	Fri Sept 16
Interim Report due:	Fri Dec 2
Final Report due:	Wed Mar 14
Final Presentations:	Fri Mar 16
Opportunities Day:	end of March (TBA)

5 Interim Project Report

5.1 Project Management Module and Deliverable

In Semester 1, all 4th year project students will attend a series of lectures on project-related topics such as project management, report writing, presentation delivery, development, plagiarism, etc. The deliverable for this module is a short report, or workplan, which serves as a basis for beginning work on the Interim Report.

This deliverable workplan for the remainder of the project will be used to compare the student's progress for the remainder of the project, so it is essential that it be a realistic representation of how you expect the project to proceed.

5.2 Interim Report Structure

The format of the interim report is fixed and its length limited to ten content pages. Apart from a cover page, abstract, full project specification, table of contents and references, there will be three content sections as follows:

1. *Introduction* (suggested 2 pages): This describes the content of the remainder of the report.
2. *Background research* (suggested 6 pages): This is a preliminary report on the background reading you have done for your project. It will usually be improved/extended in the final report.
3. *Progress Report* (suggested 2 pages): This is a very concrete report on what you have done and how you plan to complete the project. The Project Management Deliverable will feed into this section. State what you have achieved, what remains to be done, and list the milestones between here and successful project completion.

Please observe these rules carefully. They are not just optional guidelines!

5.3 Interim Report Assessment

Marks for the Interim Report will be awarded according to:

- Progress achieved to date and future schedule
- Quality and extent of background research
- Quality of workplan proposed
- Quality and presentation of the report

The Interim Report mark will be made available to the student along with comments shortly after submission, and feedback will be provided to the student from the Supervisor.

6 Final Report Assessment

The project report is a very important part of your project and its preparation and presentation should be of extremely high quality. Remember that a large portion of the marks

for your project are awarded for this report. Section 10 provides detailed guidelines and advice on the writing of this report.

In the assessment of a project, the aspects that may be taken into account include the following:

- *Problem Comprehension*: Does the student clearly grasp the problem that they are trying to solve?
- *Technical Content and Overall Approach*: Is the technical work of a high standard? Has the student used appropriate techniques in their work? Is the design compact and elegant, or is there clearly a better approach that could have been used?
- *Justification*: Does the student justify their approach? Where there are a number of options, they should be enumerated, analysed and the most appropriate one chosen.
- *Critical*: Good Science is always self-critical, and this often does not come naturally. A good project report will, in its conclusions, try to be critical of the work done, highlight its weaknesses and failings. This is not a sign of a poor project; quite the contrary. A project report that praises the work done uncritically is lacking in an important respect.
- *Completeness*: Is the project work complete? The project description makes it clear what should be achieved so the question is to what extent has the student succeeded? A project that does not achieve all its aims but provides a clear justification of why this happened is also a good result.
- *Report Quality*: Is the report well-written and presented? There should be a clear structure to the overall document and a compelling "storyline" running through each chapter. Needless to say, bad spelling, poor punctuation, illegal grammar and ill-formed sentences make a report hard to read and will result in a loss of marks.
- *Outcome of Project*: Was the project (as specified) successful? E.g. if software was produced, is the code correct? It is expected that both supervisor and PRC could be given a demonstration of your software. An implementation that is hard-to-use or buggy will not score well. The source code is also part of your final submission and this should be well-designed and laid out.

Don't get depressed looking at this list! If you could achieve all these things you would not be an undergraduate. However, it is important that you understand what a good report is and show that you are striving towards this. There is an excellent textbook by Christian Dawson on this topic that the School strongly recommends¹. It is available from the campus bookshop or from Amazon.

The format of the final report is not fixed in the way the interim report is, but its length is strictly limited. Apart

¹This sentence also demonstrates one correct way to reference external work in your report. See the References section on the last page for details of what [1] refers to.

from a cover page, project specification, abstract, table of contents and references, the recommended number of content pages² is 30 of which *at most* 10 are dedicated to background research. You should not write more than 30 pages unless you feel you have good reason to, but in any case **the maximum acceptable report length is 35 content pages**. These limits mean that you do not need to “pad out” your report. At the same time it does not make your task easy. Recall Pascal’s remark: “I am sorry for the length of my letter, but I had not the time to write a short one.”

Any implementation produced should also be submitted: see section 9.2.

Not all implementations will be scrutinised, but the factors that may be taken into account in assessing them include:

1. *Correctness*: The software should perform as described in the final report. Minor deviations are not a problem, but any major aspect described in the report should also be present in the software.
2. *Robustness*: The software should operate reliably and be not subject to frequent abnormal terminations.
3. *Code Quality*: In your years in UCD you have encountered many design and implementation heuristics. It should be apparent in the code you have written that these have guided your work.

Needless to say, a violation under point (1), that is to say a flagrant misstatement in your report of what your implementation actually does, is a very serious matter.

7 Conduct Mark

The supervisor gives a conduct mark and this is based on approach of the student towards the project. Some or all of the following criteria might apply:

1. *Approach*: How did the student approach the topic of the project
2. *Engagement*: Did the student engage well with the project
3. *Responsiveness*: Was the student responsive to the deadlines and feedback

8 Project Presentation

The final presentation takes place before an audience that will normally include the project supervisor, some members of the PRC, other staff members, postgraduates and classmates.

You will typically give a presentation and possibly a demonstration as well. The duration will typically be 15 minutes, with a further 5 minutes for questions. Please note that all students are obliged to attend throughout the days of the final presentations. Failure to attend will be noted

²In terms of the report length, appendices are also considered as content pages.

and may result in the deduction of marks. Presentations will usually be held in parallel streams.

Marks will be awarded for your presentation by the Project Review Committee based on the following criteria:

- *Technical Merit*: On a technical level, how good is the work that is being presented?
- *Organisation and Delivery*: Is the presentation well-structured and delivered clearly? There should be a logical organisation to the presentation and it should be easy to follow.
- *Handling of Questions*: You should be able to field questions on any part of your project work.

A little note on answering questions: It is vital that you listen carefully to each question and be sure that you understand it before attempting to answer. Query the questioner if you are not sure what their question is. If you feel the questioning is tangential to your project work, do not be afraid to say something like “That’s an interesting question, but my project is focused on the area of *blah...*”

8.1 4th Year Poster Session

The 4th Year Poster Session usually takes place on the same week as the final presentations, so this is a very busy week for 4th Year students. It gives you an opportunity to present your work in poster format and to get feedback from academics, postgraduates and other students. It is a requirement that you present your project at this Poster Session and failure to do so may result in the deduction of marks.

9 Submissions

9.1 Project Report Submission

Submission of all reports is via a web page. You will be provided with details of this later in the year.

Late submission will result in a sanction except where documented extenuating circumstances exist. Late receipt of a project report will normally be penalised by a loss of a full 5% for each day, or part thereof. This means that a project that is deserving of a final mark of 65% will actually be awarded only 60%, if the interim report is submitted a single day late (weekends included). Please understand from this that it is simply not acceptable to be late in the submission of any project report.

A softcopy in Portable Document Format (PDF) must be submitted. It is a simple matter to create a PDF file from a Word file, and a number of PCs with this facility will be set up at the appropriate time. No hardcopy submission is required.

9.2 Project Code Submission

Most projects involve the development of a piece of software, or some other implementation artefact. This must also be submitted, along with information on the development environment used and instructions for creating an executable. The exact details regarding how to submit your code will be

provided, but the deadline will be close to the final report deadline.

9.3 Plagiarism

Plagiarism is using other people's work in your report without acknowledging the fact that you are using it. It is most likely to occur in writing the Background chapter, as this will be based on material that you have read.

First of all, here is a tip for avoiding plagiarism. When you are actually writing a chapter, have none of the sources you are using open in front of you. Read the source material first, assimilate it, and write the description in your own words. In this way, the text you write will really be your own understanding of the area. Do provide references to your source material as well.

The School deals very firmly with instances of plagiarism, so please be very careful in this regard. It is also very naive to plagiarise in a field when your supervisor is likely to be very familiar with the source material themselves.

Partly why this is so important is that most students who plagiarise do so without realising what a serious issue it is, or what the consequences will be if it is discovered. Please be extremely wary of how you treat source material in your reports. There are a number of tools available on the web page for the projects to check your content yourself before submission.

9.4 Backups

This is so important that it cannot be overemphasized. Keep several backups of your work in various locations, so in the worst, worst case you still have something to roll back to and are not left with nothing.

10 Preparing your Project Report

Report writing is an important skill. No matter what field you are engaged in, you will almost certainly find it necessary to be able to write a clear report on your work. If you have a talent for technical writing, you will no doubt find it an easier task. However, it is a skill that can be acquired with practice and it is an essential part of your project work. Be sure to allow yourself enough time to write the report; the process generally takes *at least* two weeks.

In the following subsections I suggest an approach to take in structuring and writing your report. It is not carved in stone; feel free to adjust this to suit your own particular project/style.

10.1 Create a Report Structure

The first step is to produce a draft table of contents, showing how the entire report is to be structured into chapters, sections, and even subsections. Annotate each item with the purpose it is to serve in the overall report, and its anticipated length in pages. When you have done this ask yourself the following questions:

- Is there a logical flow through my report? If it does not flow logically at this high-level stage, it certainly will not flow well in the end either. Move sections around until you feel there is a logical thread running through the document.
- Have I written about all the important issues? Pull yourself back from the report and think about the project in general. You should not write about *everything* you did — this is a report, not a diary — but do not omit any vital sections either.
- Are the issues that I have written about important? You have probably written sections that should really be omitted. It is tough to cut out a section that you have laboured over, but dross in a report has a very negative effect on overall quality.

Once you have a sound overall structure, you can start writing sections knowing that they fit in to an overall plan. You will know how much preparatory material will have preceded each section, and you will know to what extent it is expected to lay the groundwork for later sections. You may find that you have to change the structure later in the writing of the report. As with software, the later you change the design, the more work it entails.

10.2 How Many Chapters?

The details of the structure will of course depend on the content and nature of the particular project you are working on. Generally you should break down the report into approximately six chapters. One possible template you could use is detailed below, but remember that this template is only for guidance. You may decide to merge some chapters, or have an extra core chapter. It all depends on your project and how you wish to present it.

10.2.1 Title page, Project Specification, Acknowledgments and Table of Contents

The title page should state at least the project number, project title, supervisor and your name. As with the Interim Report the full project specification should be reproduced here. In your Acknowledgments section, give credit to all the people who helped you in your project. A Table of Contents is essential, but should be produced by the word processing package you are using. The order of these is usually Title page, Project Specification, Abstract, Acknowledgments and Table of Contents.

10.2.2 The Abstract

The abstract should provide a short overview of your project that enables a reader to decide if your report is of interest to them or not. It should be concise, to-the-point and interesting. Avoid making it read like a verbose table of contents! Avoid references, jargon or acronyms, as the reader may not be familiar with them. An abstract usually contains a brief description of:

- the project and its context;

- how the project work was carried out;
- The major findings or results.

One paragraph is plenty. The main thing to remember is the principle that the abstract must be short, and a person reading it should be able to determine if they want to read more. For example, if your project involves building a compiler for Java, and a major section of your work is focussed on developing an efficient parser (rather than say code generation), make this clear in the abstract. Then a reader who is interested in efficient parsing techniques knows that your report may be of interest to them.

10.2.3 Chapter 1: Introduction

Some topics it may contain include:

- A discussion of the original aims of the project, and the modified aims if appropriate;
- The scope of the project and a general justification for the work undertaken, perhaps providing a brief background description;
- A description of the structure of the report, i.e., a road map for the reader.

10.2.4 Chapter 2: Background Research

The contents of this chapter depend on the nature of your project. If you are working on a research-oriented project, then you will present the research landscape within which your project is being conducted and consider approaches that have been adopted by other researchers. In a development project, you may describe the domain in which you are working and the technologies and programming tools you are using. Tutorial-type descriptions are never appropriate, but if you are using a specialist tool, e.g., a parser generator, it is reasonable to provide a section that describes the tool at a high level.

10.2.5 Chapters 3 and 4: The Core Chapters

These are the principal chapters of your report and their structure will vary from project to project. The aspects of your project that you will describe in these chapters include:

- A detailed account of how you approached your project, i.e. the strategy you employed. This should be at a high level, separate from design and implementation issues.
- A discussion of the design aspects of your project. Include here a discussion of interesting problems you encountered and the alternative solutions you considered.

Use the appropriate notations and formalisms in this chapter. Everything you have studied in your degree is relevant here. If there is a crucial algorithm at the centre of your project and its performance is important, attempt to provide an analysis of its complexity. If you are describing a complicated set of conditions, do not write it in English, use first-order logic. If you have performed an object-oriented design (as many of you will), use the appropriate

UML models to describe your work. Do not mindlessly produce “documentation”, but think about what you want to communicate to the reader and use the most appropriate method of doing so.

10.2.6 Chapter 5: Detailed Design and Implementation

In doing your project work, a lot of time will be spent on detailed design and implementation. The nature of programming is that it is a very time-consuming task, and even for experts a “silly” run-time error may take days to correct. In spite of this, this chapter should not be the main focus of your report. Make it clear what implementation technology you used and discuss any interesting implementation issues that arose. For example, if you were using a particular data structure that had to be optimised in a certain way to be suitable for your project, describe it in this chapter. On the other hand, if you used an obvious/standard approach, then there is no need to devote much space to it.

10.2.7 Chapter 6: Testing/Evaluation

You may decide to merge this chapter with another, but I have described it as a separate chapter as it is very important in its own right.

If the focus of your project is the development of a piece of software, then you should address the issue of how you demonstrate it to be correct. Formal proof is applicable in a small number of cases, but more commonly rigorous testing is what is required. You will not have time to really test your software in the way that industrial software is tested. However it is important to show that you have taken a methodical approach to testing and that you have tested your software in such a way that you are justified in having some confidence that it is correct. Any Software Engineering text will provide you with the basics of software testing; contact me if you want some notes on the topic.

Another type of project involves designing a heuristic or approximate solution to a challenging or ill-defined problem, e.g., to develop a junk mail filter or to mine a certain type of data from the web. In this case the precise desired behaviour of the software is hard to specify (what is junk mail anyway?), so it is more appropriate to describe how you evaluated the solution. This will involve running a number of experiments and presenting the results. As with testing, this is a complex area that you should spend some time coming to terms with. Consult [?] for some excellent advice on how to present the results of your experiments.

10.2.8 Chapter 7: Conclusions and Future Work

If you are writing this chapter bleary-eyed and caffeined-up on the day of submission, you are not going to do your project justice. This is a vital chapter in the assessment of your work. Academics, in getting the feeling for any type of report, will typically read the introduction and conclusions first. Your conclusions should not read like “I did all this stuff, it went great, and here’s other stuff someone else might do.” This chapter should cover the following areas:

- *Conclusions*: In a research-oriented project you will state the overall conclusions you have come to. In a development project there may not be a conclusion as such, so just state what has been achieved. Be very critical in this section. Describe the weaknesses of your approach and avoid making unwarranted conclusions.
- *Future Work*: Think carefully about how your work might be extended or applied to another domain. There will probably be some obvious extensions. If you are able to propose some interesting ideas that are not immediately apparent, this demonstrates that you have a clear understanding of the field.

It is good scientific style to make strong statements. If a certain statement is warranted by the results of your project, don't be afraid to make it. Strange though it may seem, a strong statement that turns out to be wrong is better than one that is vague and wishy-washy. The former can lead to a lively debate where the truth may emerge, but the latter will produce meaningless agreement, because it ultimately says nothing.

10.2.9 References

Use one consistent system for citing works in the body of your report. Several such systems are in common use in textbooks and in conference and journal papers. Ensure that any works you cite are listed in the references section, and vice versa. Word-processing packages will manage the referencing for you, and be sure to make use of this facility. It may take more time in the beginning, but at the end of the write-up it will certainly have saved you a lot of time.

In approximate decreasing order of quality, the best sources to cite are journal papers, international conference papers, national conference papers, books and web pages. Don't just supply a URL if there is an equivalent conference paper you could cite instead. Also, it strengthens your project if at least some of your references are to recently-published material.

You may instead opt for a bibliography, which is a list of material (books, papers, web resources) that you have read in preparing your project. A bibliography *must* be annotated, i.e., for each entry you must provide a paragraph summarising the work and stating why it is relevant to your project.

10.2.10 Appendices

Material that you want to include in your report, but that is not directly relevant to the main thread of your report, can be put in an appendix. Possible examples include program/code listings, detailed test results, user guides etc. In most cases, appendices are not necessary and it is only in an exceptional case that it is useful to provide a code listing.

Remember that material in the appendix counts towards report length, so do not exceed the limits defined in sections 5 and 6.

10.3 Order of Writing

The previous section dealt with one possible logical structure for your report. The order in which you write it all is

another issue. There are no fixed rules here. Some people like to write notes throughout the project, so when it comes to writing the final report, they already have a lot of material prepared. This is a very valid idea, but avoid wasting time writing very polished notes during your project work. The notes/sections you write can be quite rough, and only in the final report do you bring them up to full report quality. The reason for this is that you may have to tailor them considerably to fit the context of the report, and this will mean that much of the polishing will have gone to waste.

Assuming you have created a report structure as described in section 10.1, a good way to continue is to write the introduction in draft form. You already have a introduction from the interim report, so you can flesh this out. The reason why you write this in draft form is that you are not yet sure what you are introducing! Only when the later chapters are completed can you return and finish the introduction.

Now the Background chapter of the interim report can be revisited and improved for the final report. Again, you may find that when you write the core chapters later, some of the background work becomes irrelevant and can be removed. This may seem like wasted effort, but if it results in a tighter Background chapter, do it.

Next are the Core chapters, followed by the design and testing chapters. When these are complete, you are in a position to write your Conclusions chapter, and to return to the Introduction and Background chapters and bring them to completion. Finally, write the abstract.

The next step depends on how much time you have left. Ideally you will reach this point where you have a first full draft with at least a week to go. Proofread the report once yourself, and pass it on to other people: your supervisor of course, and anyone else who can read (part of) it and give you any sort of feedback. Take a rest yourself, so you can return to it in a day or two and re-read the report with a fresh mind.

Note that at this late stage you can only make local improvements to the report. It is too late for major overhauls, so at this point the importance of creating a good overall structure becomes clear. If you have started with a good structure, you can aim to create an excellent finished product. However if your initial structure was awkward, the final report will not read well no matter how you tweak it.

10.4 Other Advice

This section contains a number of guidelines that are worth bearing in mind when writing.

10.4.1 Continuity

You may not realise this, but a good academic paper or report, like any good novel you have read, tells a story. It is valuable to keep this in mind when you are writing your report. There should be a storyline running through your report and you should make it easy for the reader to hang on to this storyline:

- At the end of the introduction provide a short description of the layout of the remainder of the report.

- Start every chapter with a brief recapitulation of the story so far, and an overview of what the chapter is going to add.
- Finish every chapter with a summary of the material in that chapter, and state how it relates to what follows.

In the core chapters, you should take care to make absolutely clear the logical connection between the overall project design and the detailed problems you discuss. If the reader is mired in a detailed description of your solution to some intricate problem, they will be encouraged to persevere if you have clearly indicated its place in the overall project.

This continuity material may sound unnecessary and redundant, but it is useful to the reader. It may help for you to imagine that the reader is coming back to your report after a break of a few days: they will be greatly assisted by occasional reminders of what has already been said.

10.4.2 Presentation Issues

Focus on expressing your ideas clearly. Part of your report is of course its physical layout and use of diagrams. Try not to put too much time into this. Simple diagrams are fine, and avoid the use of colour unless it really contributes something in particular. Do not bother with tricks like adjusting spacing or margins or fonts in order to make your report seem bigger or smaller.

10.4.3 Managing your Supervisor at this Difficult Time

Some students unfortunately tend to avoid their supervisor in the weeks preceding the final deadline. This is perhaps because they are busy and think they need to focus on writing. One or two days before the deadline, they give their final report to their supervisor. The supervisor typically comes back to them the next day with a few minor textual corrections and says that everything is fine otherwise. The student submits their report and thinks that all is well.

The reason why the supervisor finds only few minor textual errors is because at this stage there is no point in telling the student that their report structure is deeply flawed and that one chapter should be rewritten. When the supervisor comes to assess the report (and similarly when the PRC assess and grade the report), they unfortunately have to take into account the errors and poor style that could have corrected had they been asked to review the report at an earlier stage.

In my opinion, the best way to make use of your supervisor during the writing-up period is this:

- After creating the document structure (see section 10.1), discuss it in detail with your supervisor. If you have any doubts or questions about it, hammer them out now.
- After you complete each chapter, pass it on to your supervisor and ask for feedback. Remember that they are busy, so it may take a few days for them to come back to you.

You can expect your supervisor to read at least one full draft of your report. If you are presenting it to them on a chapter-by-chapter basis, expect that they will read each chapter once.

Do take heed of your supervisor's advice. Don't present them with drafts of your report that still contain errors pointed out in an earlier version, and don't put yourself in poor light by reproducing the same errors in the remaining chapters.

10.4.4 Textual Matters

Who do you have in mind as you write your report? A good model to use is that you are writing to an educated computer scientist who is not directly involved in the field of your project. Your report is intended to be a technical document, so follow the style you see in the best scientific literature that you read.

Keep your language clear and avoid colloquialisms and abbreviations. Avoid writing in overly "academic" tones. In good academic papers you will find a simple, clear style that is easy-to-read and not overwrought. The previous sentence could also be written as: "You will find, in academic papers of good quality, a style that is at once both clear and indeed easy-to-read, without being in any sense overwrought." This style is only suitable if you are making a crowning point that you wish to emphasize heavily.

Do not use the word "I". If at some point you really wish to express a personal opinion, use a phrase like "it is the opinion of this author that...". Avoid over-using "we" as well, but don't use the passive voice all the time, or your report will be unreadable.

Avoid overuse of italicisation, underlining, bolding, or other devices for emphasis. Underlining is best avoided, as this was originally a device for informing a typesetter to use italics, and not a form of highlighting in its own right at all.

Do not place large blocks of code in your report. If you are considering putting code in your report at all, ask yourself first if an algorithm written in pseudocode is more appropriate. Any code you do present should earn its place and should be impeccable in construction and layout. Use a suitable font for code, such as *courier*, and stick to this consistently.

Pay attention to spelling, punctuation and sentence structure. Poor spelling can be very intrusive and is unforgivable given that your word-processing software surely provides a spelling checker. Poor punctuation can destroy the meaning of a sentence. If you are not sure how to use punctuation, use fewer commas rather than more. Long sentences that are difficult to write are usually also difficult to read, and may turn out in fact not to be sentences at all. If a sentence feels unwieldy, split it in two.

References

- [1] Christian Dawson, *The Essence of Computing Projects - A Student's Guide*, Pearson Education, 2000