Basic Research Skills in Computing Science

Chris Johnson

Glasgow Interactive Systems Group (GIST),
Department of Computer Science, Glasgow University,
Glasgow, G12 8QQ.

Tel: +44 141 330 6053
Fax: +44 141 330 4913
EMail:johnson@dcs.gla.ac.uk

This paper provides a basic introduction to the research skills that are necessary to complete an advanced masters or PhD in Computing Science. It starts by identifying criteria for good research practice. It then goes on to propose guidelines for the written presentation. Finally, we provide criteria for oral presentation.

This material should be read in conjunction with What is Research in Computing Science. There is also a collection of papers and advice on research and writing at CMU.

Keywords: research skills, computing science.

1. Introduction

This paper provides an introduction to 'research skills'. It presents a series of guidelines and criteria that can be used to structure your enquiries and improve the presentation of your findings. Research is, however, a **skilled** activity. Like any skill, it must be practiced. The best way to improve your research method is, therefore, to continually assess your research practice against these objectives and guidelines.

2. Research Methods

There are a bewildering diversity of research methods being used within the field of Computing Science. These include but are not limited to:

- **implementation driven research.**
  This approach progresses by iteratively building better and better systems. It has two main failings. The first is that if the system fails then you may have gained few insights into the basic research question. The failure may be more due to the limitations of the implementation than to the idea itself. The second major problem is that it can be difficult to generalise from a specific system to generic principles.

- **mathematical proof techniques.**
  This approach uses formal proofs to reason about the validity of a hypothesis given some evidence. For example, mathematical reasoning can be used to demonstrate that an algorithm can cover all possible input cases. The limitations with this approach is that the mathematical abstractions used in a proof can be too abstract or generic so that they completely ignore critical issues that must be considered during the implementation of a particular system.

- **empiricism.**

This approach follows a clearly laid out sequence of steps: hypothesis; methods; results and conclusion. Statistical measures are often use to determine whether an experiment actually supports a hypothesis. The problem with this approach is that the environment must be carefully controlled if the results of an evaluation are to be trusted. For example, comparisons between two information retrieval engines must be made by issuing predetermined queries over predetermined document sets. This does not accurately reflect the ad hoc way that users form queries `in the wild'.

- **observational studies.**
  This approach removes many of the constraints associated with empiricism by analysing the utility of a system in its eventual context of use. For example, the success or failure of a new programming language can only be assessed in this view if real programmers are trained to use the system on a real project. This is the only way to assess the impact of working pressures, deadlines and financial constraints. The limitations with this approach is that can be extremely undirected. There is no way of forcing the users to recreate particular error conditions if researchers want to observe the performance of their system in those conditions. Observational studies also depend crucially upon the analysts personal ability to read and interpret the events in the working environment. For example, many users may react differently if they know that they are being observed.

The strengths and weaknesses of these approaches are discussed in What is Research in Computing Science. The key point here is that you must determine which research paradigm is best suited to the problem in-hand. Good research often borrows methods from a number of these traditions.

A number of factors affect the success or failure of research, irrespective of the method that is being used:

- **clear problem definition.**
  If you don't understand the problem clearly then it is unlikely that you will arrive at a reasonable solution. This is an obvious remark. However, many research projects do little more than re-express a problem in a form that can then be addressed by subsequent research.

- **well defined research context.**
  There is a danger that you will `re-invent the wheel' if you do not have a good grasp of previous work in an area. Today we are well provided with web-based search engines, the University library is well equipped and staffed, the Department has many researchers who would be very willing to help you with advice and encouragement. It is critical that you spend some time consulting these sources BEFORE you embark on any research project. Otherwise you will find yourself repeating work that others have done before you. However, many research projects actively seek to replicate the results of previous investigations. This must be an explicit objective of your research and not an unfortunate product of poor planning.

- **good documentation.**
  It is critical that you spend some time documenting your daily activities when engaged on a research project. For example, it is common to find a vital URL or book reference and then lose it by not making a careful note or bookmark. Given that most research is conducted under time pressure, you only have three years for most PhDs and under a year for an MSc, you cannot afford to waste time looking for such lost resources. A good rule of thumb is that it must be possible for someone else to follow your steps from your notes alone.

- **effective time management.**
  Research is labour intensive. It takes time to find books. It can take days to complete a proof. It can take weeks to complete a program or run a set of experiments. It can take many months to write up a thesis or dissertation. You **will** run out of time if you do not carefully plan your allocation of time to each of these components. It is also important that you plan for failure. Vital pieces of your research infrastructure can and do fail. For instance, printing resources are

often stretched to breaking point in the hours before a deadline.

3. Presentation

It is impossible to do good research without good communications skills. Research is, after all, an argument between yourself and your peers to convince them of the validity of your hypothesis. If you cannot communicate effectively then you are unlikely to win the argument and you may fail to convince anyone. This section, therefore, provides guidelines to improve both written and oral presentations.

This introduction should be supported by a more complete course on communication skills.

3.1. Written Presentation

The key to successfully writing up research material is to find an appropriate level for the target audience. For example, formal methods researchers can assume a familiarity with the foundations of discrete mathematics. Software engineers can, typically, assume that their readers will have some familiarity with the concepts of iteration, selection and recursion. However, problems arise because researchers are often so familiar with their subject that they fail to explain new concepts which they consider to be `obvious'. It is, therefore, vital that you get somebody else to read your dissertation or thesis before it is submitted. This raises practical problems where the work is being examined. Giving your assessed work to another student on a taught MSc might well be viewed as encouraging plagiarism. At the very least, you must allow time for your supervisor to read your final draft before submission.

Even if you successfully establish the correct level of discussion for your audience, you must still determine which material is necessary for the reader's understanding of your argument. For example, in formal theses you must decide how far to go in the presentation of supporting lemmas and proofs. Some of this material can be included in the running text, some of it can be dropped to an appendix, some of it can simply be cited in other work. Similarly, in an implementation driven project it is seldom necessary to include the entire listing in the body of the dissertation. Key concepts can be illustrated by appropriate procedures and functions but most of the code can be relegated to an appendix.

It is possible to identify a number of further guidelines:

- **follow grammatical rules.**
  If you are unsure about what constitutes good grammar then you should spend some time reading a recognised reference work.

- **follow conventions on font selection.**
  Fonts are designed for a purpose, that purpose is often indicate by their name. For example, New Century Schoolbook was designed for school text books. The Times family of fonts were developed for the Times newspaper. It, therefore, follows that you should select a font that is appropriate for the purpose that you have in mind. Fonts can be divided into two categories: serif and sans serif. Serif fonts, such as Times or Bookman, have small additions to the ends of each letter that help to lead your eye along the line. In contrast, sans serif fonts such as Helvetica or Geneva lack these additional visual cues. They appear to be simpler and because your eye is not led along the line, each word will stand out as a discrete unit rather than part of a line. The easiest way to see this effect is to compare two copies of a document; one in Times Roman and the other in Helvetica. The practical outcome from all of this is that headings should, typically, be presented in a sans serif font. You want the heading to stand out as a discrete unit, distinct from the rest of the text. Running text should be in a serif font because you want to help the reader's eyes to move along the line. **Do not use large numbers of fonts without a carefully considered reason. Do not use bizarre and obscure fonts that may not easily be printed on**

other people's machines

- **use a recognised reference format**
  Every citation in the body of the thesis must be supported by a corresponding reference. The reference must provide enough information so that another researcher would be able to find the paper or book with minimal effort. For an article you must include the author's name, the article title, the journal name, the volume and edition, the page numbers and the year of publication. For a book you must include the author's name, the book title, the publishers name and town of publication, and the year of publication. For a chapter you must include the author's name, the chapter's title, the editors' names and the book's title, the page numbers of the chapter, the publishers name and town of publication, and the year of publication. There are examples in my publication's list. Most examiners will search your references looking for the important works in the field. They will also look for their own publications. **Make sure these are listed**.

- **use a spell checker and proof read your thesis.**
  Remember, it is not your examiners job to identify typos and spelling mistakes. If they spend too much time doing this then your marks will almost certainly be affected. If your thesis is in a more formal area of Computer Science, remember whenever possible to use a type checker.

- **follow HTML guidelines**
  Ultimately, many research works are now being published on the web. As this becomes an increasingly common means of communication, it is worth making the effort to transform your thesis into html format. This does not simply involve a straight transformation from the printed document to its web counterpart. You will end up with a huge file that The Sun Microsystems web style guide is a good reference source.

3.2 Oral Presentation

The guidelines for oral presentations are similar to those for written work. As before, the two critical concepts are **level** and **relevance**.

It is critical that researchers present their material at a level of detail that their audience can follow. The same level of detail is NOT appropriate for both written and oral presentations. A common mistake is simply to increase the point size of your dissertation and then attempt to use this as lecture material. This does not work because you will, typically, only have 20-30 minutes to present your findings. This means that you have to condense written material that may take several hours to read, into a format that can be effectively presented in a matter of minutes. You must apply Occam's razor: only present material that is essential to make your case. **Do not attempt to increase the amount of material you can cover by talking fast or by glossing over more basic material that the audience will need to understand your findings.** It is better to be convincing in a small subset of your work that to be glib and superficial about a larger body of work.

Relevance is also important in spoken presentations. It is good practice to prepare slides on background material and actually ask the audience whether or not they would need this introduction in order to place your work in context. For instance, in a presentation about graphical user interface design you might, typically assume that the audience understood the term `direct manipulation'. However, you should be prepared to offer a definition and an example if asked to do so. Similarly, it is good practice to pause after every key point and ask the audience whether they are have any questions about the concepts that you are introducing.

The following guidelines and check list are derived from the Committee of Vice Chancellor's and Principles' Guidance on Effective Learning and Teaching in Higher Education.

Opening

1. Does your opening gain the group's attention?

2. Does it establish rapport with the group?
3. Does it indicate what you intend to explain?

The Key Points

1. Are your key points clearly expressed?
2. Are your examples apt and interesting?
3. Are your qualifications of the key points clearly expressed?
4. Is each key point summarised?
5. Are the summaries clear?
6. Are the beginnings and ends of the key points clearly indicated?

The Summary

1. Does the summary bring together the main points?
2. Are your conclusions clearly stated?
3. Do you come to an effective stop?

Presentation

1. Can the group hear and see you?
2. Do you use eye contact to involve but not threaten?
3. Do you use audio/visual techniques effectively?
4. Are you fluent verbally?
5. Is your vocabulary appropriate for the group?
6. Do you make use of pauses and silences?
7. Do you vary your intonation?
8. Is the organisation of your material clear?
9. Do you avoid vagueness and ambiguities?
10. Is the presentation as interesting as you can make it?

4. Conclusion

This paper has provided a basic introduction to the research skills that are necessary to complete an advanced masters or PhD in Computing Science. Pointers have been given to related resources on the web. Ultimately, however, research skills can only be gained through practice in the application of the guidelines that are presented in this paper.