CS2309 CS Research Methodology

Introduction

Lee Wee Sun
School of Computing
National University of Singapore
leews@comp.nus.edu.sg

Semester 1, 2010/11

Why learn research?

Competence

MIT has long believed that professional competence is best fostered by coupling teaching with research and by focusing attention on real-world problems – *MIT web site*

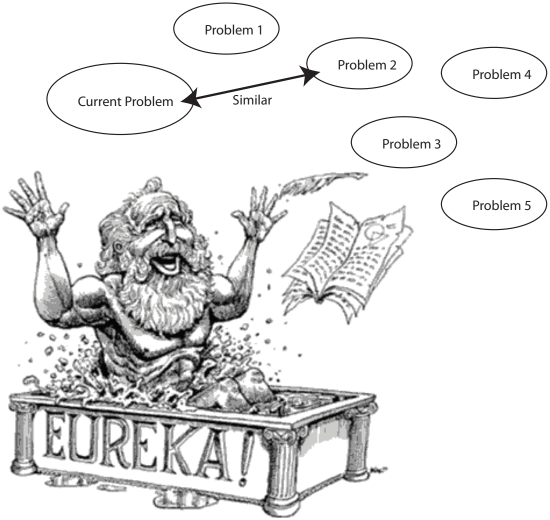
- In the real world, many problems that you face do not have known fixed solution.
- The process of finding solutions to unknown problem is research.
- Doing research is one of the best way to develop competence for dealing with real technical problems.

Learn how to:

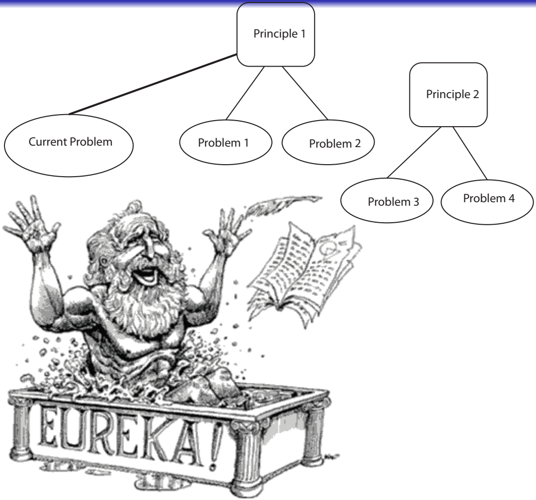
- Formulate the problem
  - What is the real issue?
  - Will it have impact if I solve it?
- Search for solution
  - How to find a solution quickly?
- Validate solution
  - How can I be confident that the solution will work?

Valuable high level skills regardless of what you do!

Can creativity be learned?



Can creativity be learned?



Learn principles: easier to connect to principles.
Get into the habit of asking question and trying to connect things.

Aim

Teach useful *principles* for

- Problem modeling
- Developing solutions
- Convincing yourself and others of the solution.

Process

- Work through commonly used principles.
- Study and discuss examples of how the principles help.
- Study landmark research papers as case studies.
- Project to gain some hands-on experience.

Objectives

By the end of the course, you should have

- Used abstraction to simplify and model several application problems.
- Analysed several systems using commonly used principles.
- Developed several solutions using algorithmic and heuristic problem solving principles.
- Proved a few results using some common proof techniques.
- Designed a few experiments without obvious flaws.
- Perform statistical hypothesis testing to validate several hypotheses.
- Analyzed a few impactful works and understood why they had impact.
- Gained experience in applying some of the above in a project.

Grading

- Project Presentation 10%
- Project Report/Achievement 25%
- Paper Presentation 10%
- Paper Comments 10%
- Homework 10%
- Exam 30%
- Participation 5%

Paper Reading

- Some of tutorials will be used for presenting papers.
- Each student must present at least once this semester.
- Each session we will have one or two papers to read along the same topic.
- Presenters will cover the papers, relevant background and related ideas on the topic.
- All students will need to submit a one page comment for each paper, describing your opinion and analysis (except presenters for the particular papers)

Project

Research project in a small group (two or three)

- Formulate a problem
- Solve the problem
- Validate solution
- Write up and present

Have common theme to increase class discussion.

- This year: *Social Computing*.
- See IVLE Forum for discussion.

Can undergraduates do good research?

Yes, many have done wonderful research projects in their UROP and FYP here.

`http://www.comp.nus.edu.sg/undergraduates/urop_samples.html`

Great Research

A lot of you here have enough brains to do good work. But great work requires something more ...

- Work on important problems.
- Great scientists keep important problems in their mind
  - Go after it whenever a line of attack appears
  - This is what luck means!

Great Research

You cannot do without **hard work**!

Newton

If others would think as hard as I did, then they would get similar results.

Hamming

Given two people of approximately the same ability and one person who works ten percent more than the other, the latter will more than twice outproduce the former. The more you know, the more you learn; the more you learn, the more you can do; the more you can do, the more the opportunity - it is very much like compound interest.

Edison

Genius is 99% perspiration and 1% inspiration.

Great Research

Knowing a lot is not enough.

Need great emotional commitment.

- Spend a lot of time thinking about the problem.
- Creativity comes from subconcious and manifest itself only if you spend a lot of time thinking about a problem - often days and nights.

However,

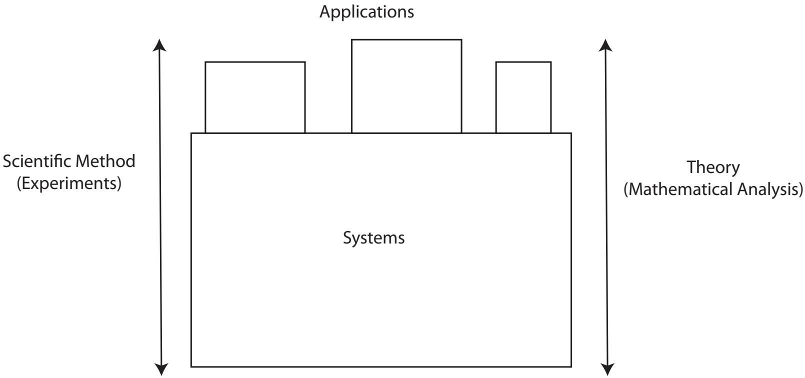
- Only working on important problem is unproductive.
- Work on easier related problems or subproblems.
- Make your problem important.

See Hamming's talk at

`http://www.cs.virginia.edu/~robins/YouAndYourResearch.html`

(This week's reading.)

Types of Research



For convenience, we categorize types of research into *theory*, *systems*, and *application* research.

Theory

- Abstract out a model from a real-world phenomena.
- Use mathematical analysis. Prove theorems.
- Provide understanding.
- Sometimes provide useful algorithms, design or architectural principles.

Theory

Example: Halting Problem

Seminal paper by Alan Turing in 1936.

- Turing machine as model for human computation.
- Showed that there are problems that Turing machines cannot solve.
- Negative answer to Hilbert's Entscheidungsproblem: Give an algorithm that takes a formal language and a mathematical statement in that language and outputs whether the statement is true or false.

Systems

- Build a large system to realize a vision.
- Often large group. Research often depends on parts built by others.
- Need ideas, then build system to demonstrate the ideas.
- Generates useful techniques: virtual memory, garbage collection, packet networks, relational databases, transactions, GUIs, ...
- Often generates useful side products, e.g. ARPANET, the predecessor to the Internet

Systems

Example: Relational Database

Relational model defined by Codd around 1970.

- Early to mid 1970's several groups built relational database systems to demonstrate that it can be made practical
- E.g. System R from IBM, Ingres from Berkeley
- Side products: SQL, Ingres system

Applications

- Lots of application areas for computing
- Logistics, computer vision, natural language processing, games, etc.
- Contributions are often new models or new algorithms.
- Need domain knowledge.
- Validation method like those of other scientific domains: mathematical analysis or experiments.

Applications

Example: Biological sequence alignment

- Given a protein sequence, find a similar protein sequence.
- Abstract the problem into string comparison problem.
- Develop/apply string comparison algorithms for efficient solution.

What research topic should I work on?

Do what you are interested in!

Considerations

What is the outcome if you are successful?

- Would it improve the lives of people?
  - ARPANET $\rightarrow$ Internet.
- Would it generate wealth, for yourselves or for others?
  - Google $\rightarrow$ multi-billion dollar business.
- Would it improve our understanding of the natural world, or of the mathematical world?
  - NP-completeness $\rightarrow$ mathematical, physical limitations.

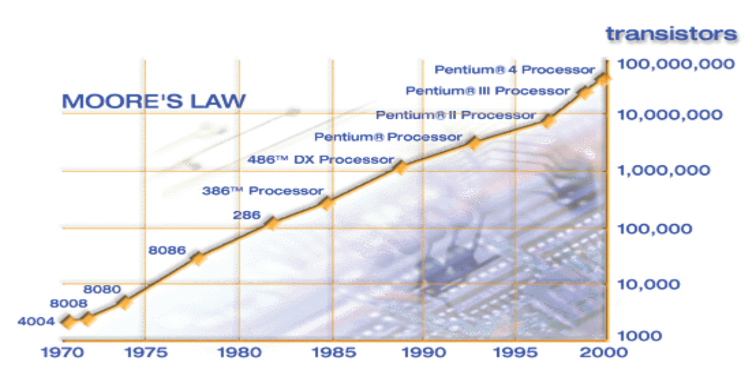
- Would it allow things that cannot previously be done to become possible?
  - Public key cryptography $\rightarrow$ secret communication without prior exchange of keys.
- Would it make things 10 times, or 100 times better?
  - Interior point methods give more speedup than Moore's Law since invention.

Trends

One way to gain some insights into what is likely to be useful is to look at trends. Moore's law is an interesting example.

Moore's Law

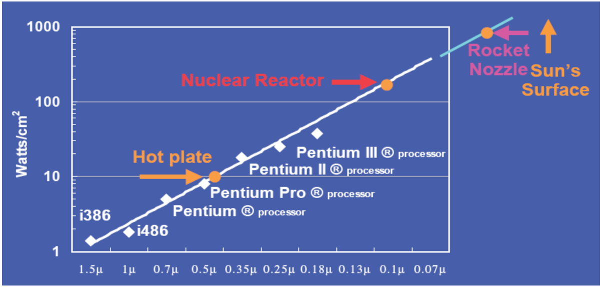
Moore's law predicts an exponential increase in computing power[a]

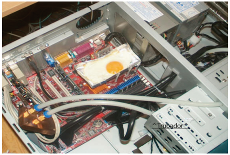


[a] Image from `http://www.intel.com/research/silicon/mooreslaw.htm`

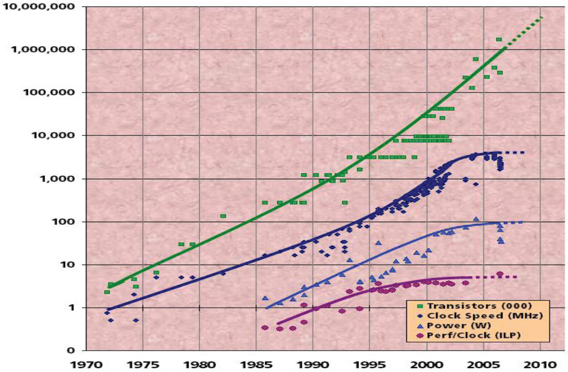
- Parallel computing systems had difficulty gaining traction - not cost effective.
- Failure rate of parallel computer companies is 100%: Convex, Encore, MasPar, NCUBE, Kendall Square Research, Sequent (Silicon Graphics), Transputer, Thinking Machines, etc.
- Fifth Generation Computer Systems project in the 1980s, partially dedicated to developing highly parallel computer architectures, failed.

- Interestingly, we are now at an important crossroad of this paralled computing trend.
- Single processors are getting too hot to keep improving at the same exponential rate.





- Moore's law still holds for the doubling of transistor density in integrated circuits - switched to increasing the number of processor cores instead.



- Improved performance in future will have to come from parallelism instead of increasing clock speed.
- Looking at the trend, perhaps, parallel computing and power aware computing are currently good systems research topics.

Strengths

Do what you are good at!

Explore

- Don't get too carried away with cost benefit analysis
  - Particularly when you are just starting out
- Follow your interest
- Explore
  - Of course, exploit when opportunity comes along
- Don't know what will be useful in future
- The more you know, the likelier things will connect

Example: Steve Jobs and Apple Typography

http://www.youtube.com/watch?v=D1R-jKKp3NA

Getting Started

Don't just sit around and read about other people's work.
Get your hands dirty!

- Every paper you read, think about how YOU can extend the work
  - Small extensions generally easier and are a good way to start.
- Every situation you encounter, think about how YOU can resolve it
- Try things out to see how they work.
- Talk to people to find out what is happening.

References

Required Reading

1. You and Your Research, *Richard Hamming*, Transcription of the Bell Communications Research Colloquium Seminar 7 March 1986. http://www.cs.virginia.edu/~robins/YouAndYourResearch.html