

# Introduction To Computer Science

In this section you will get an overview of some research areas and higher level courses in Computer Science.





# Some Areas Of Study And Research In Computer Science

- Human-Computer Interaction
- Computer Graphics
- Information Visualization
- Databases
- Computer theory
- Computer networking and distributed systems
- Artificial Intelligence
- Computer Vision
- Software Engineering
- Computer Security
- Games programming

This list provides only a brief introduction to the different areas of Computer Science and is far from comprehensive:  
For a more updated list of research areas: <http://www.cpsc.ucalgary.ca/Research/>

Calendar (courses):

- <http://www.ucalgary.ca/pubs/calendar/current/computer-science.html>
- <http://www.ucalgary.ca/pubs/calendar/current/software-engineering.html>

# Some Areas Of Study And Research In Computer Science

- **Human-Computer Interaction**



- **Computer Graphics**

- Information Visualization



- Databases

- Computer theory

- Computer networking

- **Artificial Intelligence**

- **Computer Vision**

- **Software Engineering**



- **Computer Security**

- **Games programming**

Full disclosure:  
I've done work in  
these areas

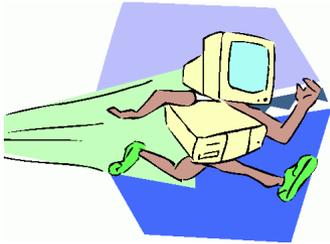
This list provides only a brief introduction to the different areas of Computer Science and is far from comprehensive:  
For a more updated list of research areas: <http://www.cpsc.ucalgary.ca/Research/>

Calendar (courses):

- <http://www.ucalgary.ca/pubs/calendar/current/computer-science.html>
- <http://www.ucalgary.ca/pubs/calendar/current/software-engineering.html>

# Human-Computer Interaction (HCI)

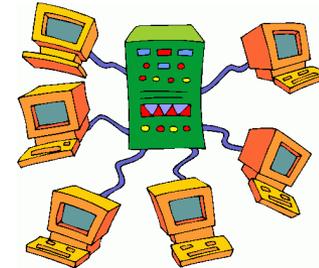
- Most of Computer Science deals with the ‘technical’ side of computers.



Run computers faster!



Make computers store more information!!

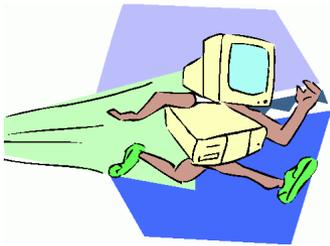


Increase the networking capabilities of computers!!!

- These technical issues (and others) are all very important but something is still missing...

# Human-Computer Interaction

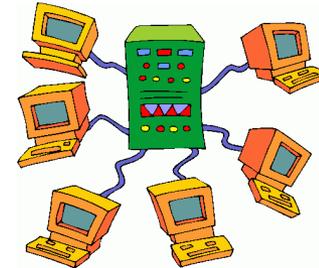
- Most of Computer Science deals with the ‘technical’ side of computers.



Run computers faster!



Make computers store more information!!



Increase the networking capabilities of computers!!!

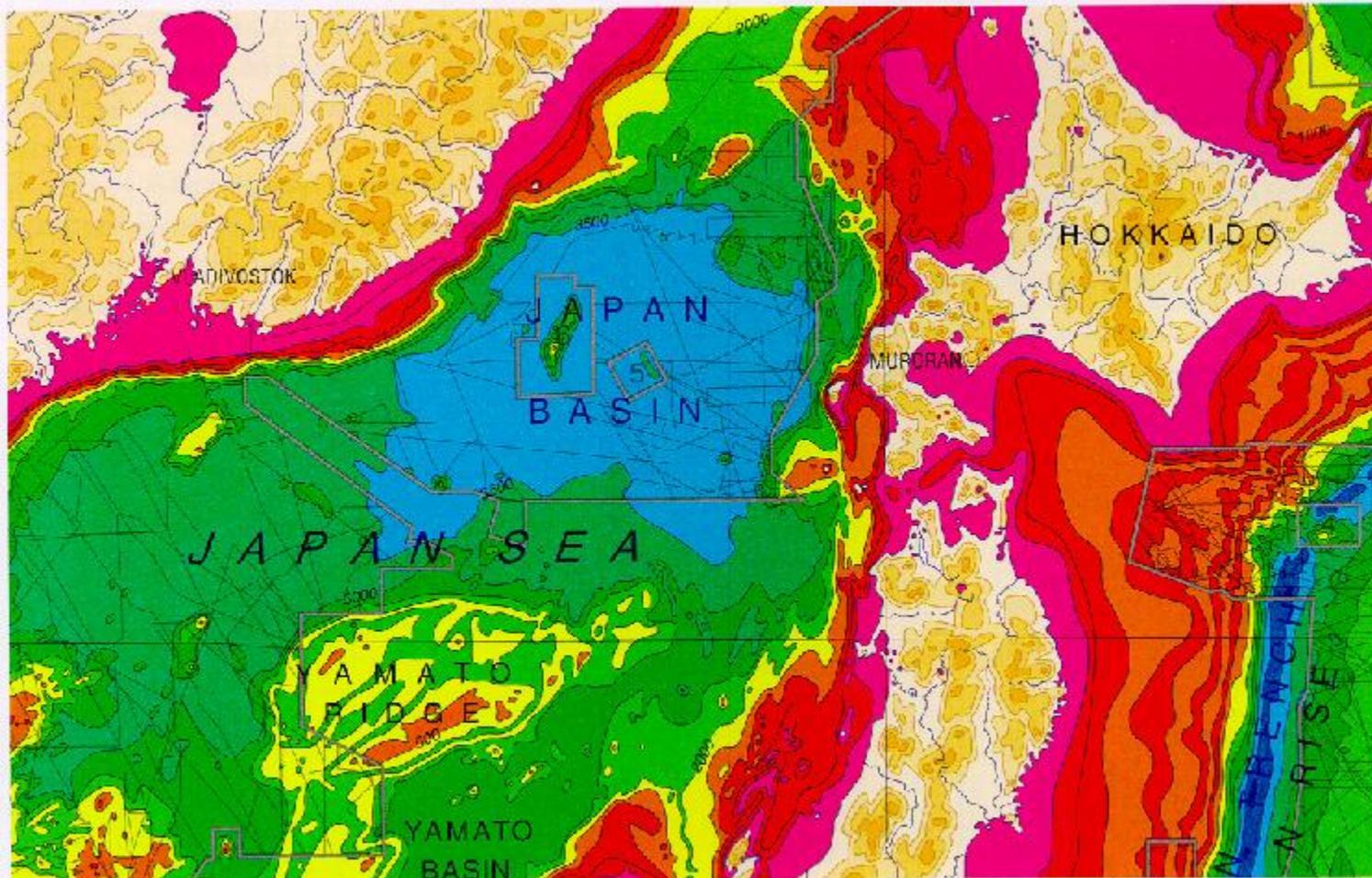
- These technical issues (and others) are all very important but something is still missing...

# Human-Computer Interaction

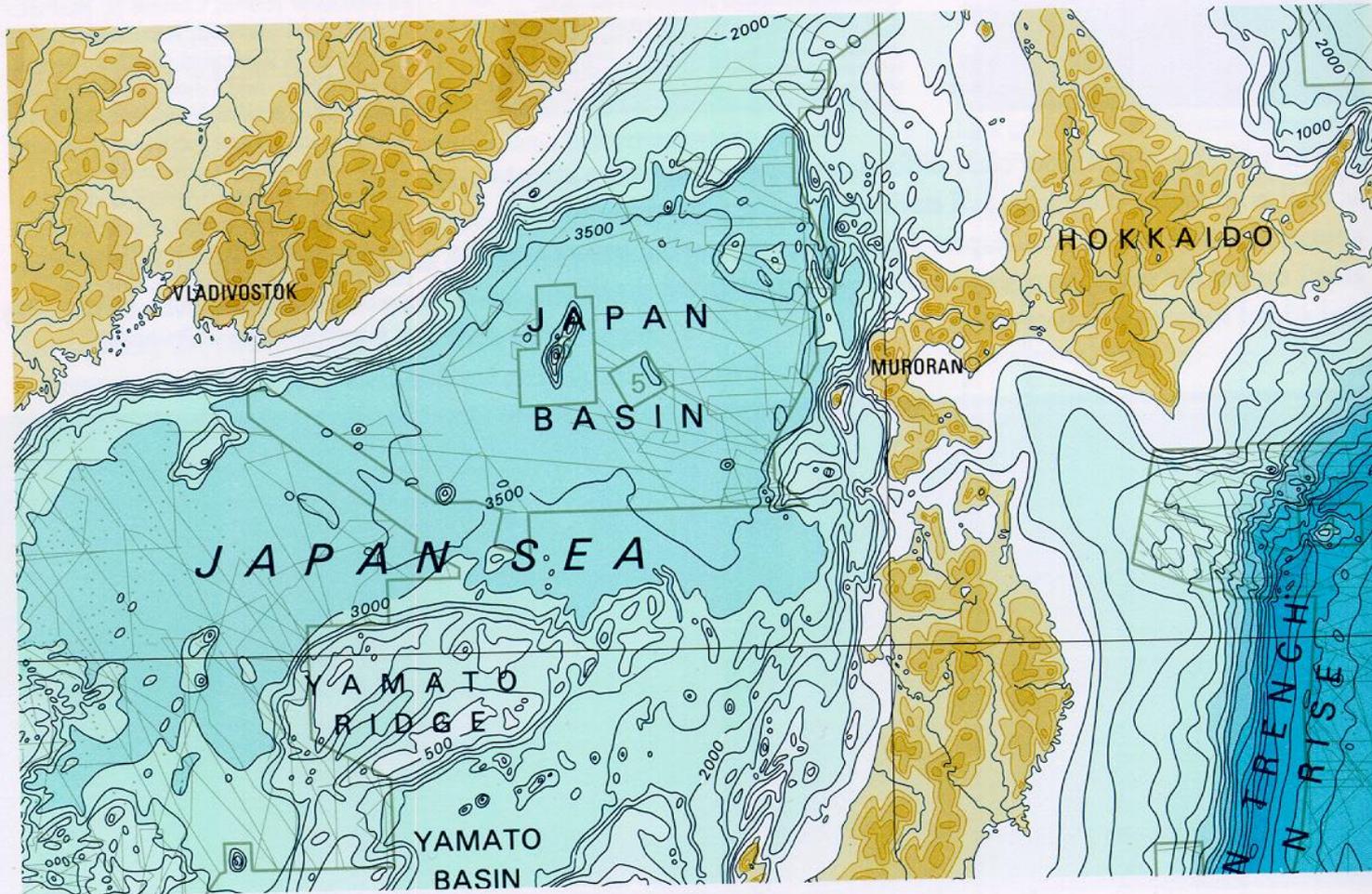
- ...but don't forget about the other side of the relationship.
- No matter how powerful the computer and how well written is the software, if the user can't figure out how it works then the system is useless.
- Software should be written to make it as easy as possible for the user to complete their task. (Don't make it any harder than it has to be).
- This is just common sense and should/is always taken into account when writing software?

*Common sense?...come on!*

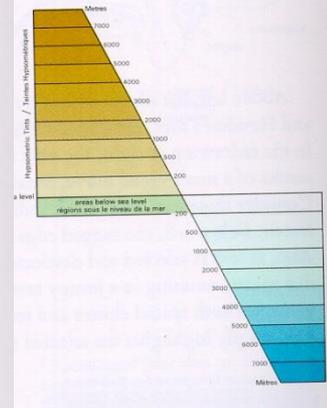
# Human-Computer Interaction: Not Just Common Sense Information (Using Color)



# Human-Computer Interaction: Not Just Common Sense Information (Using Color)



General Bathymetric Chart of the Oceans,  
International Hydrographic Organization  
(Ottawa, Canada, 5th edition, 1984). 5.06.



# What Is Human-Computer Interaction?

~~Difficult to use~~

Easy to use

Or at least easier to use

# How Can This Be Done?

- Many techniques have been developed.
  - Some may have already been covered (heuristics)
- One other technique: simple but effective (user-centered design)
  - Basic principle: getting users involved in the design process from the beginning (rather than building the system and then getting feedback afterwards which is the traditional approach).
  - Many benefits:
    - Cost reduction: The further along the software development process the harder it is to make changes.

## Paper sketches

A hand-drawn paper sketch of a payment options screen. The title is "PAYMENT OPTIONS". Below the title, there is a box labeled "Amount Due \$" and another box labeled "Method of Payment". Under "Method of Payment", there are several options: "Cash \$", "Visa \$", "MC \$", "Bank \$", and "Debit \$". To the right of these options is a numeric keypad with digits 1-9, 0, and a decimal point. At the bottom of the screen, there are two buttons: "Make Payment" and "Receipt Data".

## Complete software

A screenshot of a complete software interface for a stroller. The interface is divided into several sections. At the top left, there is a "What to do" section with the instruction "Touch a different color, or scan another item." and a small icon of a person's face. To the right of this is a "What you selected" section for a "JPG Stroller" intended for children between 1-3 years old, priced at \$98. Below this, there are three color options: "Green" (selected with a checked box), "Blue", and "Red (out of stock)". Below the color options is a table with columns for "Item", "Style", and "Cost". The table shows "JPG Stroller" in "Green" style for a cost of "98.00". Below the table, there is a "tax: 6.98" and a "Total: \$104.98". At the bottom, there is an "All done?" section with three buttons: "Place your order", "Print this list", and "Throw this list away".

- Users may also provide many unexpected insights

# HCI: Higher-Level Courses

- CPSC 481: Human-Computer Interaction I
- CPSC 581: Human-Computer Interaction II
- (Related: Human-Robot Interaction)
  - CPSC 599.65—Robot head-based interaction
  - CPSC 599.62—Advanced topics in human-computer and human-robot interaction
  - CPSC 599.17—Human-robot interaction

# Computer Graphics

- Concerned with producing and manipulating images on the computer.



Gran Turismo © Sony

# Computer Graphics: Issues

- How to make the images look “real”?



From <http://klamath.stanford.edu/~aaa/>

# Computer Graphics: Common Misconception

- It's about creating the programs that produce the realistic images and animations (not using existing programs like PhotoShop ©).



# Computer Graphics: Still A Long Way To Go

- “Even though modeling and rendering in computer graphics have been improved tremendously in the past 35 years, we are still not at the point where we can model automatically, a tiger swimming in the river in all it’s glorious details.” <sup>1</sup>



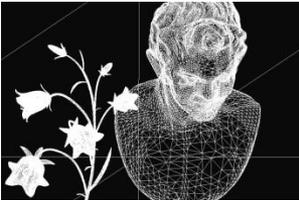
<sup>1</sup> From “The Tiger Experience” by Alain Fournier at the University of British Columbia

# Graphics: Some Areas

- Animations



- Modeling



Xin Liu

- Rendering



Xin Liu

- Imaging

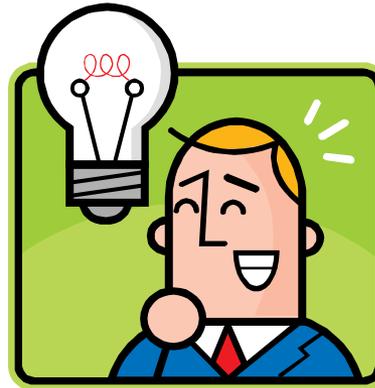
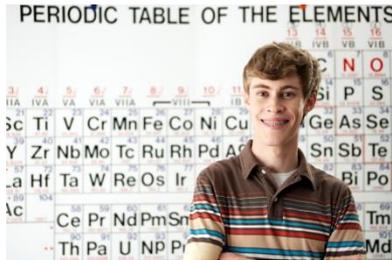


# Graphics: Higher-Level Courses

- CPSC 453: Introduction to computer graphics
- CPSC 587: Fundamentals of computer animation
- CPSC 589: Modeling for computer graphics
- CPSC 591: Rendering

# Artificial Intelligence

- Trying to build technology that appears to be ‘intelligent’
- Intelligence: What makes a person smart?



For more information:

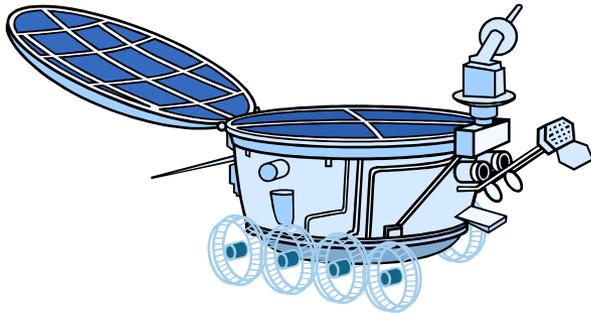
- <http://pages.cpsc.ucalgary.ca/~jacob/AI/>
- <http://pages.cpsc.ucalgary.ca/~denzinger/>
- <http://pages.cpsc.ucalgary.ca/~kremer>

# Artificial Intelligence: Some Areas

- Machine learning
- Expert systems
- Neural networks

# Machine Learning

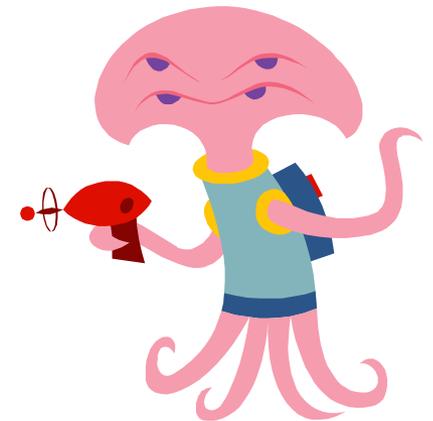
- The focus is on designing a computer that has the ability to learn and adapt to new situations (rather than just apply a fixed set of rules).



Pre-set rules: terrain



New scenario: life form encountered



Pre-set rules: terrain



# Expert Systems

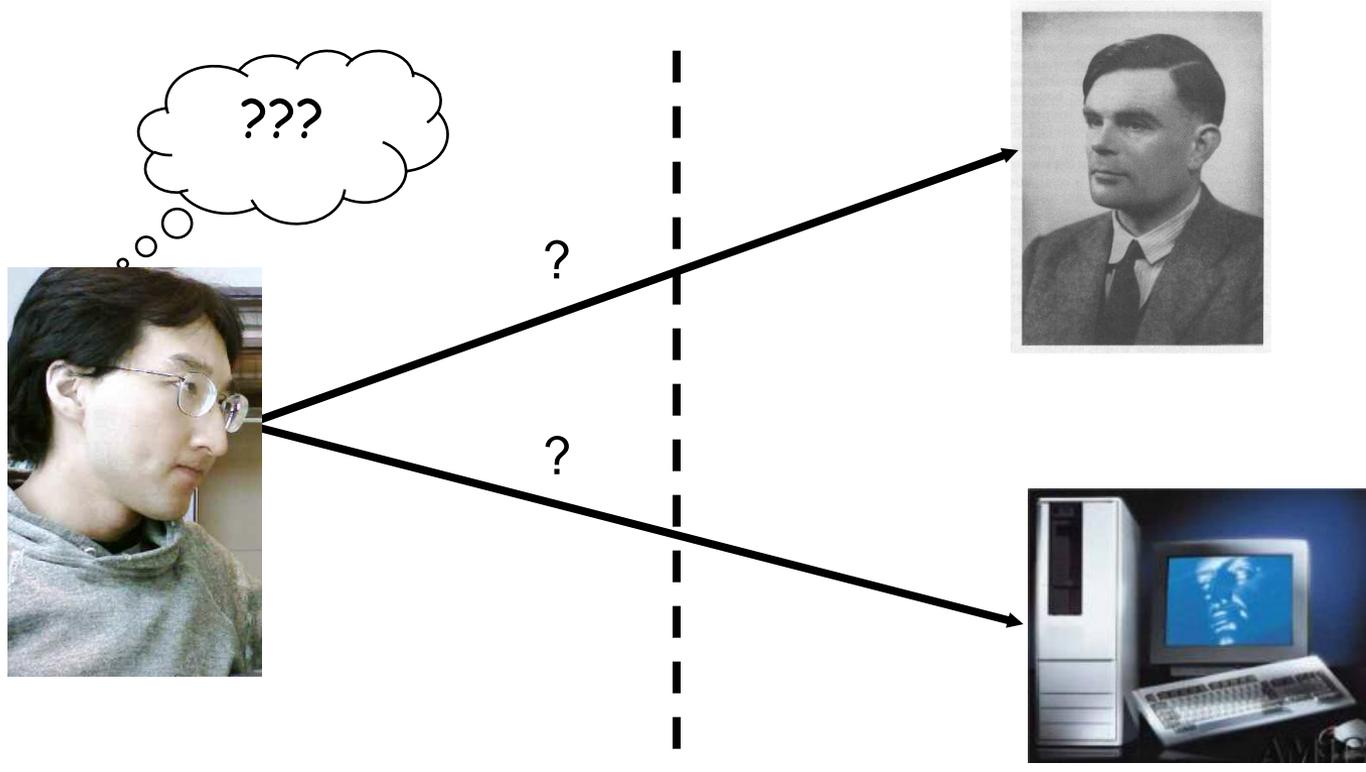
- The focus is on capturing the knowledge of a human expert as a set of rules stored in a database.
- The expert system can then answer questions, diagnose problems and guide decision making.
- Example applications: medicine, computer repair

# Neural Networks

- The focus is on building structures that function the way that neurons (and their connections in the brain) function.
- (Simplified overview):
  - Neurons take electrical pulses as input and send electrical pulses as output.
  - A required level of input is required before the output is ‘fired’.
- This approach has been applied to problems which involve pattern recognition ( e.g., visual, voice).

# Artificial Intelligence: Mission Accomplished?

- How do we know we have a "smart machine"?
  - The Turing test



# An Artificial Intelligence Won't Be Created In The Foreseeable Future

- Much work still needs to be done: Turing Test not yet passed



Photo from [www.startrek.com](http://www.startrek.com) © Paramount

# Artificial Intelligence: Higher-Level Courses

- CPSC 433: Artificial Intelligence
- CPSC 565: Emergent computing
- CPSC 567: Foundations of multi-agent systems
- CPSC 568: Agent communications

# Computer Vision

- The focus is on interpreting and understanding visual information.



For more information:

<http://pages.cpsc.ucalgary.ca/~boyd/pmwiki/pmwiki.php?n=Main.Research>

<http://people.ucalgary.ca/~jparker/>

# Computer Vision: Some Areas

- Recognition



Image-based searches



Identification of malignant cells (mockup)

- Restoration

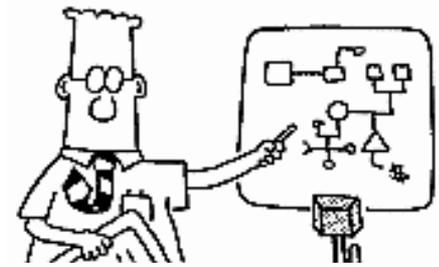


Removing imperfections such as blurring

# Computer Vision: Higher-Level Courses

- CPSC 535: Introduction to image analysis and computer vision
- CPSC 583: Introduction to information visualization

# Software Engineering



Dilbert © United Features  
Syndicate

- Concerned with employing systematic ways of producing good software on time and within budget.
- A typical person can only hold  $\sim 7$  concepts in their mind at a time.
  - A typical computer program consists of more than 7 'parts'.
- Consequently mechanisms for dealing with this complexity are needed.
  - Top down approach is one way: break a large (hard to conceive) problem into smaller more manageable parts.

For more information:

[http://www.cpsc.ucalgary.ca/cpsc\\_research/areas/evolutionary](http://www.cpsc.ucalgary.ca/cpsc_research/areas/evolutionary)

James Tam

# Software Engineering (2): Techniques

- Agile development
- Design patterns

# Agile Programming

- The focus is on reducing risk by producing a new ‘iteration’ of the software in a short period of time (~1 – 4 weeks).
- The project is then evaluated.
  - The emphasis is on real time and face-to-face communication between developers over written documentation.
  - Everyone associated with the project is brought together: developers, software testers, project managers and end users.
  - Benefit: reduced development time with fewer misunderstandings.
- Contrast with traditional development: formal processes are followed such as heavily documenting program code.
  - The code is written by a programmer.
  - Documentation is the way that others understand how the code works.
  - The client may be periodically be asked to “sign-off” on the software.

# Agile Programming (2)

- Traditional approaches work well for extremely large projects that require a high degree of reliability.
- Agile programming works well for smaller (although still large) projects where having a shorter development time is crucial.

# Design Patterns

- A design pattern: a way of creating software that has been shown to be sound under a number of different contexts.
- Design patterns are a way of documenting successful past approaches
  - Top down design: although not one of the formally recognized designed patterns it shares some similarities to those approaches.

# Software Engineering: Higher-Level Courses

- Software Engineering 301 Analysis and Design of Large-Scale Software I (required for all CPSC majors)
- Software Engineering 401 Analysis and Design of Large-Scale Software II
- Software Engineering 403 Software Development in Teams and Organizations
- Software Engineering 437 Software Testing
- Software Engineering 471 Software Requirements Engineering
- Software Engineering 511 Software Process and Project Management
- Software Engineering 513 Web-Based Systems
- Software Engineering 515 Agile Software Engineering

# Software Engineering: Higher-Level Courses (2)

- Software Engineering 521 Software Reliability and Software Quality
- Software Engineering 523 Formal Methods
- Software Engineering 533 Software Performance Evaluation
- Software Engineering 541 Fundamentals of Software Evolution and Reuse

# Computer Security

- It can involve the creation of malicious software ('malware')



Spam generators



- Purpose: learn about how malicious software is created and distributed.
- Goal: develop countermeasures to protect computer systems



Virus software



Spyware

For more information:

<http://icis.cpsc.ucalgary.ca/>

# Some Approaches To Computer Security

- As just demonstrated, understanding ‘how things work’ is one key component to designing more secure systems.
  - e.g., Creating viruses and other malware in order to create better defenses against them.
- But also the ‘human’ factor must be considered: some security experts think that many security breaches are due to user actions not technical flaws.
  - But this may require more than just standard ‘security workshops’.

# Cryptography

- As may have already been mentioned (depends on assignments), cryptography can play an important role in security.
  - Transmitting and storing sensitive information.
  - Cryptography involves the development of new and better approaches for encoding sensitive data (to make unauthorized access harder).

# Computer Security: Higher-Level Courses

- CPSC 329: Explorations in information security and privacy
- CPSC 418: Introduction to Cryptography
- CPSC 525: Principles of computer security
- CPSC 527: Computer viruses and malware
- CPSC 528: Spam and spyware
- CPSC 530: Information theoretic security

# Games Programming

- Pulls together many areas of Computer Science
- The University of Calgary was the first Canadian university to offer this area of study.

<< Warning!!! >>

Blatant  
advertisement

<< Warning!!! >>



“Scarface: The World is Yours” © Radical Entertainment

Sound byte: © “The Simpsons” Fox

# Computer Games: Higher-Level Courses

- CPSC 585: Games programming

# After This Section You Should Know

- What are some areas of Computer Science
- What does each area entail
- Some of the sub-areas, techniques employed or issues associated with each area of computer science