Introduction To Computer Science

In this section you will get an overview of some areas of Computer Science.

James Tam

Introduction To Computer Science

•What is Computer Science?

James Tam

Introduction To Computer Science

•What is Computer Science?

James Tam

Introduction To Computer Science

•Computer Science is about problem solving

Graphics

Interactive displays

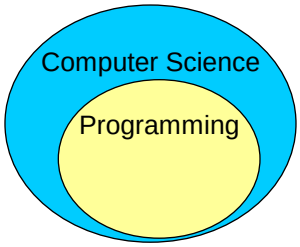
Robotics: acceptance of domesticated robots

Artificial Intelligence
FIFA © Electronic Arts.

James Tam

Computer Science Is Not The Same As Computer Programming

•Computer Science does require the creation of computer programs ('programming') but goes beyond that.



James Tam

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: http://www.cpsc.ucalgary.ca/Research/

James Tam

Some Areas Of Study And Research In Computer Science

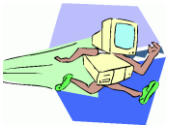
- **Human-Computer Interaction**
- **Computer Graphics**
- Information Visualization
- Databases
- Computer theory
- Computer networking
- **Artificial Intelligence**
- **Computer Vision**
- **Software Engineering**
- **Computer Security**
- **Games programming**

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

James Tam

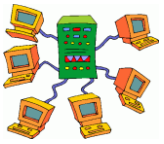
Human-Computer Interaction (HCI)

- 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!!!

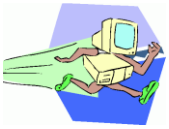
- These technical issues (and others) are all very important but something is still missing...

For more information: http://ilab.cpsc.ucalgary.ca/

James Tam

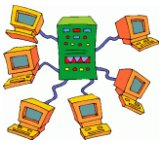
Human-**Computer** Interaction

- 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...

For more information: http://ilab.cpsc.ucalgary.ca/

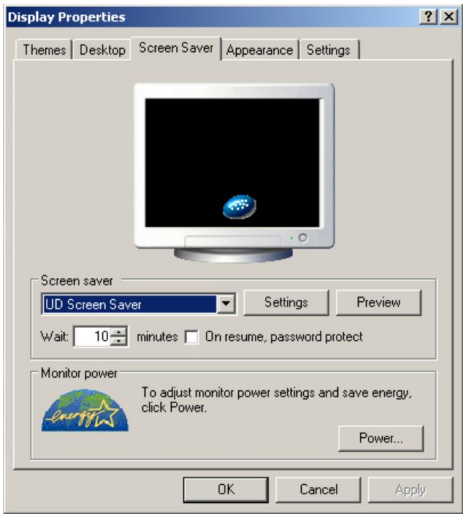
James Tam

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 of the program 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?

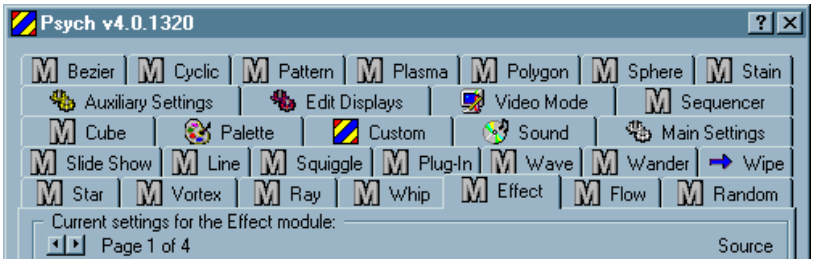
James Tam

Human-Computer Interaction: Not Just Common Sense Information



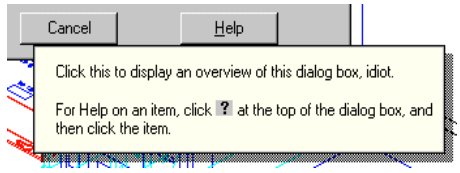
James Tam

Human-Computer Interaction: Not Just Common Sense Information (2)

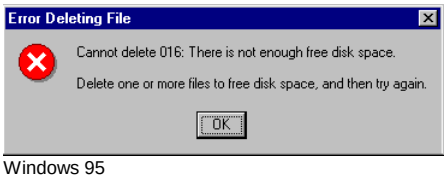


James Tam

Human-Computer Interaction: Not Just Common Sense Information (3)



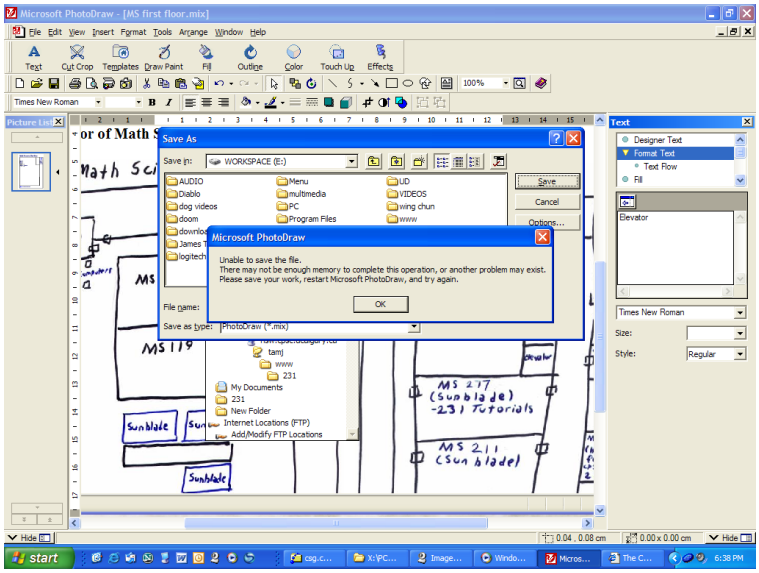
AutoCAD Mechanical



Windows 95

James Tam

Human-Computer Interaction: Not Just Common Sense Information (4)



James Tam

What Is Human-Computer Interaction?

~~Difficult to use~~

Easy to use

Or at least easier to use

James Tam

Determining Requirements For Software

- Requirements are typically a list of 'features' or operations that the software performs.
  - E.g., for a word processor it could include saving, printing, spell checking etc.
- While having the proper functionality is important (fulfilling the requirements) it's not sufficient.
  - Although a program might include a particular feature if users cannot find or figure out how to use the feature then it's useless.

James Tam

Ways Of Including The 'Human' In The Development Process

- Get in touch with real people who will be potential users of your system.
- Spend time with them discussing how the system might fit in to their work.
- Learn about the user's tasks:
  - Articulate concrete, detailed examples of tasks they currently complete or those that they want to complete (ones that they want to do but can't do

James Tam

Computer Graphics

- Concerned with producing images on the computer.

Gran Turismo © Sony

For more information: http://jungle.cpsc.ucalgary.ca/

James Tam

Computer Graphics: Issues

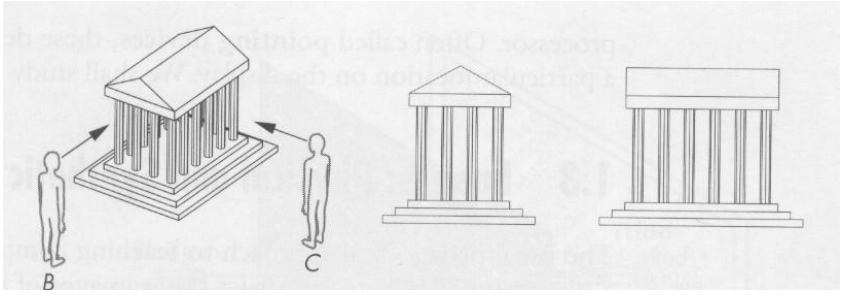
•How to make the images look "real"?

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

James Tam

Computer Graphics: Highly Mathematical

•Highly mathematical



James Tam

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." [1]

[1] From "The Tiger Experience" by Alain Fournier at the University of British Columbia

James Tam

Artificial Intelligence

- Trying to build technology that appears to be 'intelligent'
- What makes a person smart?

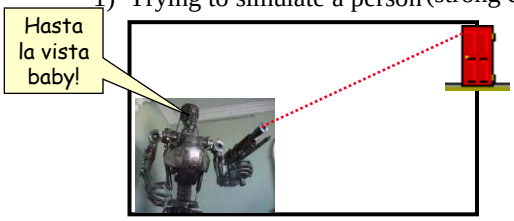
"*We don't truly understand what intelligence is, but we hope we know it when we see it*." – Kenneth Hoganson (Concepts in Computing): Jones and Bartleett 2008)

For more information:
http://pages.cpsc.ucalgary.ca/~jacob/AI/
http://pages.cpsc.ucalgary.ca/~denzinge/
http://pages.cpsc.ucalgary.ca/~kremer

James Tam

Artificial Intelligence (2)

- Approaches:
  1) Trying to simulate a person (strong equivalence)

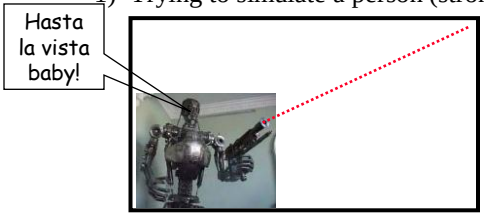


  2) Trying to simulate what the person can do

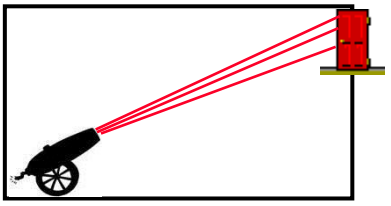
James Tam

Artificial Intelligence (2)

- Approaches:
  1) Trying to simulate a person (strong equivalence)



  2) Trying to simulate what the person can do (weak equivalence)



James Tam

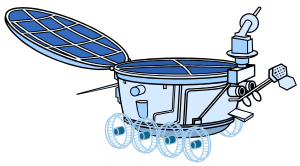
Fields Of Artificial Intelligence

- Machine learning
- Experts systems
- Neural networks
- Fuzzy logic

James Tam

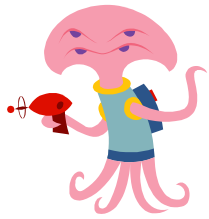
Machine Learning

- 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

James Tam

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

James Tam

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 and 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).

James Tam

Fuzzy Logic

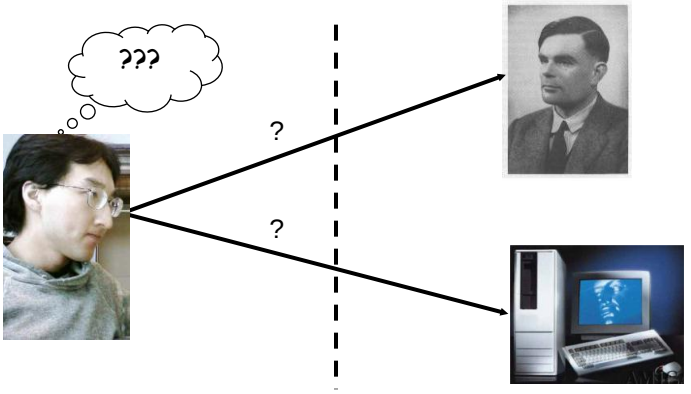
- People can make decisions when faced with uncertainty.
- The standard logic of computer programs (true/false) cannot be easily applied when relationships can be applied with only a degree of probability.
  - Standard computer program:
    if X then Y
    e.g., if temperature >= 50 then decrease temperature

- Fuzzy logic is reasoning with probabilities
  - Fuzzy logic programs:
    If close enough to X then Y
    E.g., if temperature is too hot then make temperature less hot

James Tam

Artificial Intelligence: Mission Accomplished?

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



James Tam

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

•Much work still needs to be done

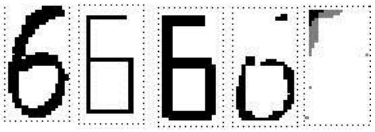
Photo from www.startrek.com © Paramount

James Tam

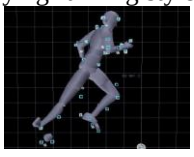
Computer Vision

•The focus is on interpreting and understanding visual information.

- Hand writing recognition: six?



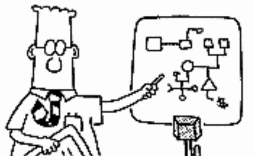
- Analyzing digital video: studying running styles (i.e., not just still images)



For more information:
http://pages.cpsc.ucalgary.ca/~boyd/pmwiki/pmwiki.php?n=Main.Research

James Tam

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 ~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 break a large (hard to conceive) problem into smaller more manageable parts.

For more information:
http://www.cpsc.ucalgary.ca/cpsc_research/areas/evolutionary

James Tam

Software Engineering (2): Techniques

- Extreme programming
- Agile development
- Design patterns

James Tam

Extreme Programming

- The focus is on developing prototypes very quickly with extensive testing and user communication.
- With the traditional approach to software development where specifications (what the software is supposed to do) is determined at the start and fixed throughout the project ("sign offs" may occur).
- With extreme programming specifications can and will change.
  - (It's argued that it's impossible to correctly envision all the issues associated with a large project at the onset).
  - There is however greater risk that the software will run into 'dead ends' and it has to be redesigned.
- Consequently with extreme programming changes occur to the software to adapt to things like changing client requirements.

James Tam

Agile Programming

- Related to extreme 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.

James Tam

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.

James Tam

Design Patterns

- A design pattern: a way of implementing and part of the software that has been shown to be been sound under a number of different contexts.
- Design patterns are a way of documenting past approaches to a problem that have shown be successful.

James Tam

Computer Security

•Involves 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

Spyware

Virus software

For more information:
http://icis.cpsc.ucalgary.ca/

James Tam

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

For more information: http://www.cpsc.ucalgary.ca/undergrad/courses_progression/concentration?conc=game

James Tam