COSC 122
Computer Fluency

Privacy and Security

Dr. Firas Moosvi

Acknowledgement: Original slides provided courtesy of Dr. Lawrence and Dr. Abdallah Mohamed.

Final Exam Information

1) You should aim to be at the exam room at least 10 minutes before the start of the exam.

2) Once you arrive, your ID will be checked and you'll be randomly assigned a seat.

3) You will NOT be able to submit your work on paper , the exam will be on PrairieLearn and you will need probably need Visual Studio Code setup to do well.

4) Once you submit your exam on PrairieLearn and your Work, you may leave.

Final Exam Information

1) 50% of the exam will be similar to the multiple choice questions you've seen on Tests so far.

2) 50% of the exam will be similar to the coding questions you've seen on Test 4 (and will see on Test 5).

3) The content that we will cover will include everything covered on Tests 1-5 including a small handful of questions on Today's content.

Important: Bonus Test 5 Date Change

Because the Term ends on Thursday Dec. 8th, Bonus Test 5 needs to be on Wednesday Dec. 7th

Key Points

1) Privacy involves ensuring personal information is used and distributed according to a person's wishes.

2) Security encompasses the various ways for ensuring privacy and protecting digital data.

3) Security includes user identification, access privileges, and protocols and encryption.

4) Encryption encodes text so that only the intended receiver can understand it.

Privacy

Privacy is the right of people to choose freely under what circumstances and to what extent they will reveal themselves, their attitude, and their behavior to others.

Information technology threatens privacy due to the ease of storing, copying, and exchanging digital information that is collected from a variety of sources (government, business, etc.).

As users of services, we are often forced or must "voluntary disclose", private information that we trust the organizations will keep secure and not distribute.

Although there are numerous rules and regulations for privacy, they are **not consistent** across all countries and **cannot always be rigorously enforced.**

Privacy: Whose Information Is It?

An interesting question about privacy relates to who "owns" information in a transaction or exchange.

For instance, when you **buy groceries**, does the grocery store have the right to the information about what you purchased?

- This is **valuable information** to the merchant as they can spot trends that help in marketing and inventory management.

Beware: If you have any sort of **membership card**, your purchase information can be maintained across visits to get a profile of your purchases.

- Merchants can also use your credit or debit card information.
- Most organizations now must disclose how they will use the information and give you the **right to "opt out"**.

A Privacy Success Story
So Long Tele-marketers!

Before: The telemarketing industry's "self-policing" mechanism required individuals to write a letter or make an on-line payment to **stop telemarketing calls**. Individuals received numerous, unwanted calls.

Solution: The United States government set up the **Do-Not-Call** List. Anyone on the list cannot be called by a tele-marketer without incurring a fine.

Result: There are over 80,000,000 households on the list and the telemarketing industry has largely collapsed.

In Canada: The government has passed legislation creating a Do-Not-Call list similar to the United States.

- https://www.lnnte-dncl.gc.ca

Privacy on the Internet

The Internet is ***not*** **an anonymous** communication system.

- **User ids, cookies, and IP addresses** can be used to track communications and interactions.
- Any **interaction** with a web site can be **logged and recorded**.
- **Email** travels (and may be logged) by numerous servers.

Privacy can only be guarded with adequate security and knowledge.

You must assume that anything you do on the web will become public.

Privacy Breaker: The Cookie

A ***cookie*** is a small file stored on your computer by your browser by a web site that you visit.

A cookie file allows **a site to identify you** between visits by storing information such as your user id.

Cookies can be **abused by advertisers** who store them on your computer whenever you visit a site they have ads on. They can then use the user id in your cookie to detect when you visit other sites that they provide advertising for.

Browsers now give you the option of **disabling cookies** on a per site, individual request, or overall basis.

Your Digital Footprints

Your **Internet activities are recorded** in a variety of places which results in a large digital footprint:

- **Browsers store**: Browsing history and cache, form data, cookies, etc.
  - Learn how to delete them <u>or</u> use Incognito mode or Anonymizer.
- **ISP stores**: Some traffic information, bandwidth usage, potentially logs of sites visited
- **Cellphone companies store**: History of calls, cell phone towers used, call detail records, text message content/detail, and IP information. Some of this information is stored for over a year and is available without a warrant.

Identity Theft

Identity theft is the crime of posing as someone else for fraudulent purposes.

It is too easy to **get personal information** for others:

- from **spam** email or bogus web sites
- from **security breaches** in registered databases
- from **accidental release** on the Internet
- from **paper records** including discarded documents

Identity theft is a growing problem because most **financial transactions** are entirely automatic. Once you have the key identifying fields for a person, ***a system assumes you are that person*** and **no manual verification is performed**.

Protecting Your Identity

It may sound paranoid, but in today's digital society, your identity is your most important asset and must be protected:

- Ensure your **computer security** including anti-virus and software is up-to-date.
- Only use **trusted software**, **email**, and **web services**.
- Be wary of **scams** that are "**Too good to be true**!"
- Chose **strong passwords** and keep them safe.
- **Shred documents** that contain personal and financial data.
- **Do not trust** an organization or person unless you have evidence that you should do so.

Security

Security is the act of keeping precious data safe and only **accessible to the correct people**.

- Security is a way of enforcing privacy in digital systems.

There are many different security technologies. In general, security involves several things:

(1) User identification - verify system user is who they say they are

(2) Access privileges - only allow user to access data they have the privilege (or right) to access or update.

(3) Security or encryption protocol - stores or transmits data in such a way that only users with the correct access privileges can use it.

(1) *User Identification*

A system performs ***user identification*** to determine if the user is who they claim to be.

Technologies for user identification:

- **User id and a password**
  - **The most common form** of user identification.
  - An ***authentication system*** is used to verify the user id and password is correct.
- **Biometrics** - finger printing, voice recognition, eye scans
- **Digital access cards** and keys

Creating Good Passwords

Your password is your only form of defense against other users accessing your data and private information.

- It is crucial to select a good one because **there are techniques to "crack" passwords**, especially weak ones.

Cracking passwords:

- **Directed guessing** - use common words, names, birth dates, and other information known about the user.
- **Brute force** - try all possible character sequences to find the password (usually limited by denying access after a while)

Good passwords have **at least 6 characters** with a mixture of upper and lower case letters, numbers, and punctuation.

- It should not contain components of dictionary words or personal information.

Changing Passwords

Passwords should be changed periodically.

Although managing passwords for many different systems is cumbersome, **using a single password for everything is risky**.

A good idea is to recycle passwords by rotating through a few or making slight changes to existing ones.

Question: Why can the administrator not tell me my forgotten password?

- Answer: Passwords are encrypted when stored on the computer to prevent the administrator (and others) from knowing it.
- Administrators are only allowed to reset a password.

(2) *Access Privileges*

Access privileges limit access to data and software functions based on the rights assigned to the user.

The ***access control system*** verifies a user has access to the given resource before allowing them to use it.

On shared machines, your user id provides you access to some files and programs. However, you cannot typically access the files and directories of other users unless they allow you to.

Three common access privileges:

- ***read*** - can read file contents
- ***write*** - can update file contents or delete entire file
- ***execute*** - can run a program or enter a directory

These access privileges may be specified on a **per user** basis, to ***groups*** of users, or to all users (**public**).

(3) *Encryption And Decryption*

An ***encryption system (protocol)*** converts data into a form that cannot be understood by anyone but the intended user.

- ***Encryption*** transforms a data representation so it is no longer understandable to users without the decryption key.
- ***Decryption*** converts an encrypted data representation into its original form, usually using a key or private information.

Cleartext or ***plaintext*** is the information before encryption.

Cipher text is the information in encrypted form.

A ***cryptosystem*** is a combination of encryption and decryption methods.

A ***one-way cipher*** is an encryption system that cannot be easily reversed (used for passwords).

Caesar Cipher

The ***Caesar cipher*** was used by Julius Caesar to encrypt messages sent to his generals.

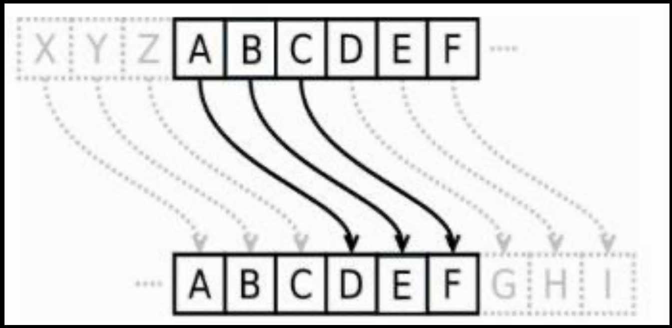
- Encryption Algorithm: Shift each letter over K places, wrapping around to the start of the alphabet as necessary.
- Decryption Algorithm: Go back K letter places in the alphabet, wrapping as necessary.

Example (K=3):

- `Plain text  =` **ABCDEFGHIJKLMNOPQRSTUVWXYZ**
- `Cipher text =` **DEFGHIJKLMNOPQRSTUVWXYZABC**

Example (K=3):

- Plain text = `HELLO WORLD.`
- Cipher text = `KHOOR ZRUOG.`



Question: Pick a partner and exchange a short encrypted message (then decrypt).

Backing Up a Personal Computer

What to backup:

- All personal data including documents, pictures, and music.
- Software settings such as Internet favorites.
- Do not backup operating system or programs as they can be re-installed from source CDs.

How to backup:

- **Simple**: Use a duplicate device such as a USB key or extra hard drive and copy files to it periodically.
- **Offsite**: Burn a CD or DVD with files and store in another place.
- **Online**: Use cloud services (iCloud, Dropbox, Box.com).
- **Sophisticated**: Install and configure backup software that regularly saves data to another drive or CD/DVD.

Backup

Question: The last time I backed up the important files on my computer or laptop was…

A) Last week

B) Last month

C) Last semester

D) Last year

E) Never … do you mean the computer can lose my files?

Conclusion

Preserving our ***privacy*** is especially important in our digital world because of the amount of information collected and the simplicity that it can be exchanged.

Security protocols and systems are designed to restrict access to systems and data to the appropriate individuals.

- Security involves user identification (authentication system), access privileges (access control system), and encryption.
- We must use good passwords to protect our privacy.

Various encryption protocols provide data security. RSA public encryption is a strong encryption scheme.

We must backup our data and system in addition to securing it.

Objectives

- Discuss some issues with maintaining privacy in a digital world.
- Define cookie and explain how it can invade your privacy.
- Define identity theft and list some precautions to avoid it.
- Define security and list three components of security.
- Define: user identification, access privilege, authentication system, access control system
- Define: encryption system, encrypt, decrypt, plain text, cipher text, cryptosystem, one-way cipher
- Draw a diagram and explain how encryption/decryption works.
- Be able to encode and decode a Caesar cipher.
- Explain the key idea of public (RSA) key encryption.
- Define: system backup, redundancy

COSC 122

Computer Fluency

Limits of Computation

Dr. Firas Moosvi

Acknowledgement: Original slides provided courtesy of Dr. Lawrence and Dr. Abdallah Mohamed.

Key Points

1) Computers can demonstrate “artificial intelligence” but cannot yet mimic human creativity.

2) Game trees and search strategies are used to create the intelligence in games.

3) Scientists use big-Oh notation to analyze and compare the performance of algorithms.

4) There exists some problems where there is no efficient solution or no solution at all.

Do Computers have Limits?

We have seen that the computer gets it **power** by being able to perform **simple operations very fast**.

By combining these simple operations into larger software programs, a computer can perform complex tasks.

Two interesting questions:

- 1) Can computers do anything (everything)?
- 2) Can computers behave like humans?

Can Computers Think?

The ***Turing Test***

Source: www.alanturing.net

If the judge cannot identify the computer, the computer can be said to be intelligent.

Can Computers Think?

Alan Turing posed the ***Turing Test*** to evaluate if a computer can mimic a human. The Turing Test:

- Two identical rooms connected electronically to a judge who can type questions directed to the occupant of either room. A human being occupies one room, and the other contains a computer. The judge's goal is to decide, based on the questions asked and the answers received, which room contains the computer. If after a reasonable period of time the judge cannot decide for certain, the computer can be said to be intelligent.
- The computer is intelligent if it acts enough like a human to deceive the judge.

The test does not define thinking, intelligence, awareness or focus on any specific ability.

Can Computers Think? (2)

Computers have come closer to passing the test but not quite.

Computer advances:

- Better understanding and parsing of **natural language (ELIZA)**
- Recognize **semantics** in language and communication
- **Translate** to and from natural language realistically
- Improved computational power

Work to do:

- Computational power will go beyond that of the human brain in 50 years.
- Outstanding challenge is modeling **complexity and intelligence** in software.

Acting Intelligently?

Eliza is a system developed by MIT researcher Joseph Weizenbaum to carry on a conversation as though she were a psychotherapist.

- The software used word clues for answers but did not understand the meaning so it was not intelligent.

Example:

User: I'm depressed.

Doctor: Why are you depressed?

User: My mother is not speaking to me.

Doctor: Tell me about your mother.

User: She doesn't want me to major in engineering.

Doctor: No?

User: No, she wants me to go into medicine.

Artificial Intelligence

Artificial intelligence (***AI***) refers to the ability of a computer to **mimic human intelligence** in certain situations.

To exhibit intelligence, the computer has to "**understand**" a complex situation and **reason** well enough to act on its understanding.

One example of AI is computer intelligence in playing games such as **chess** and **checkers**.

Game Intelligence

For a computer to play a game against a human opponent, it must make **intelligent decisions on** its moves.

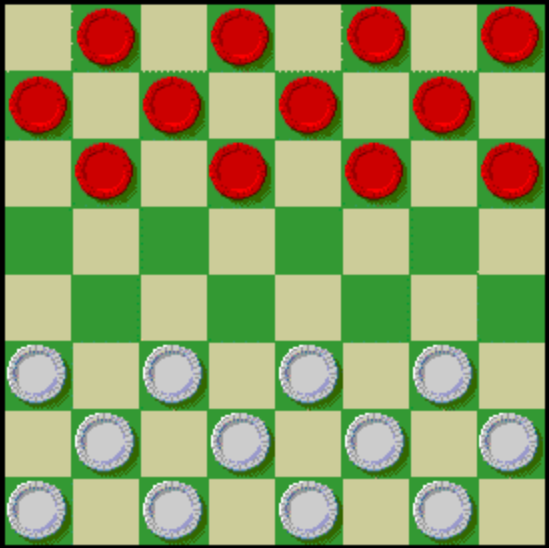
Strategy games such as checkers and chess have been targeted games for computing "artificial intelligence".

Even video games require the computer to determine **strategies**, even though the decision making is less complex.

- This includes games such as role-playing games and strategy/conquest games.

Checkers

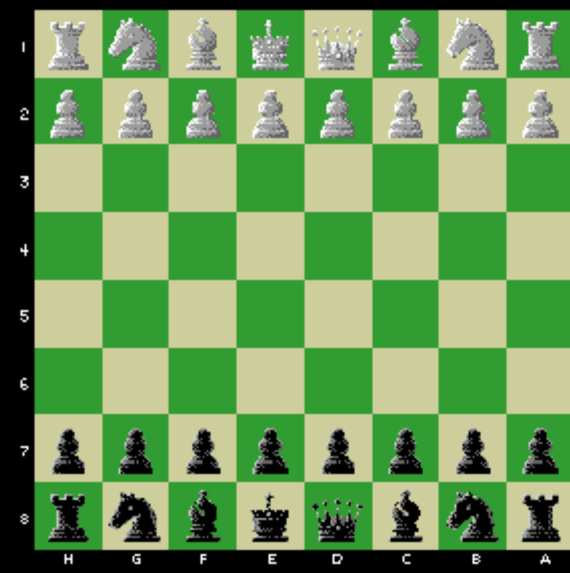
The world champion is a program called **Chinook** created by researchers at the University of Alberta, Canada.



- http://www.cs.ualberta.ca/~chinook
  - Get crushed by Chinook if you choose
- They also have a research group called GAMES (Game- playing, Analytical methods, Minimax search, and Empirical Studies)
  - http://www.cs.ualberta.ca/~games
- Checkers was solved in 2005. Perfect play by both sides leads to a draw. There are 500 billion billion (5×10^{20}) positions.
- Group is working on poker players as well.

Chess

A computer program does not have world champion status, but defeated world champion Gary Kasparov in a regulation match in 1997.



Deep Blue

- Developed by IBM at a cost of millions of dollars.
  - Powered by a RS/6000 massively parallel mainframe.
  - Can evaluate 200 million board positions a second.
  - http://www.research.ibm.com/deepblue/home/html/b.html

Jeopardy

IBM's Watson AI program competed on Jeopardy! in February 2011 against champions Ken Jennings and Brad Rutter.

- Watson: $77,147   Jennings: $24,000 Rutter: $21,000

Watson is a specialized program with a huge, self-contained database that **parses English, formulates queries to database, and filters and selects correct answer**.

- Database has 200 million unstructured pages.
- Watson consisted of 2,800 computers and terabytes of memory.
- Applied to medicine, banking, and research.

"Final Jeopardy!" question it got wrong in category U.S. Cities: *Its largest airport is named for a World War II hero, its second largest for a World War II battle.*

A Simple Game
Tic-Tac-Toe

A good way to look at structures and algorithms that are capable of playing games of pure skill is by examining a simple game like Tic-Tac-Toe (also called x's and o's).

Tic-Tac-Toe

- A game of pure skill
  - No element of chance
- Can program Tic-Tac-Toe by "looking" for forced moves, traps, and patterns.
  - Careful case by case analysis
    - Can be done because of the relatively **few cases possible**

Game Playing
Mini-max Strategy

The majority of game playing systems employ something called a ***mini-max strategy***.

- The basic idea in a mini-max strategy is that you determine a move which ***maximizes*** your potential to win the game and ***minimizes*** your opponent's potential to win the game.

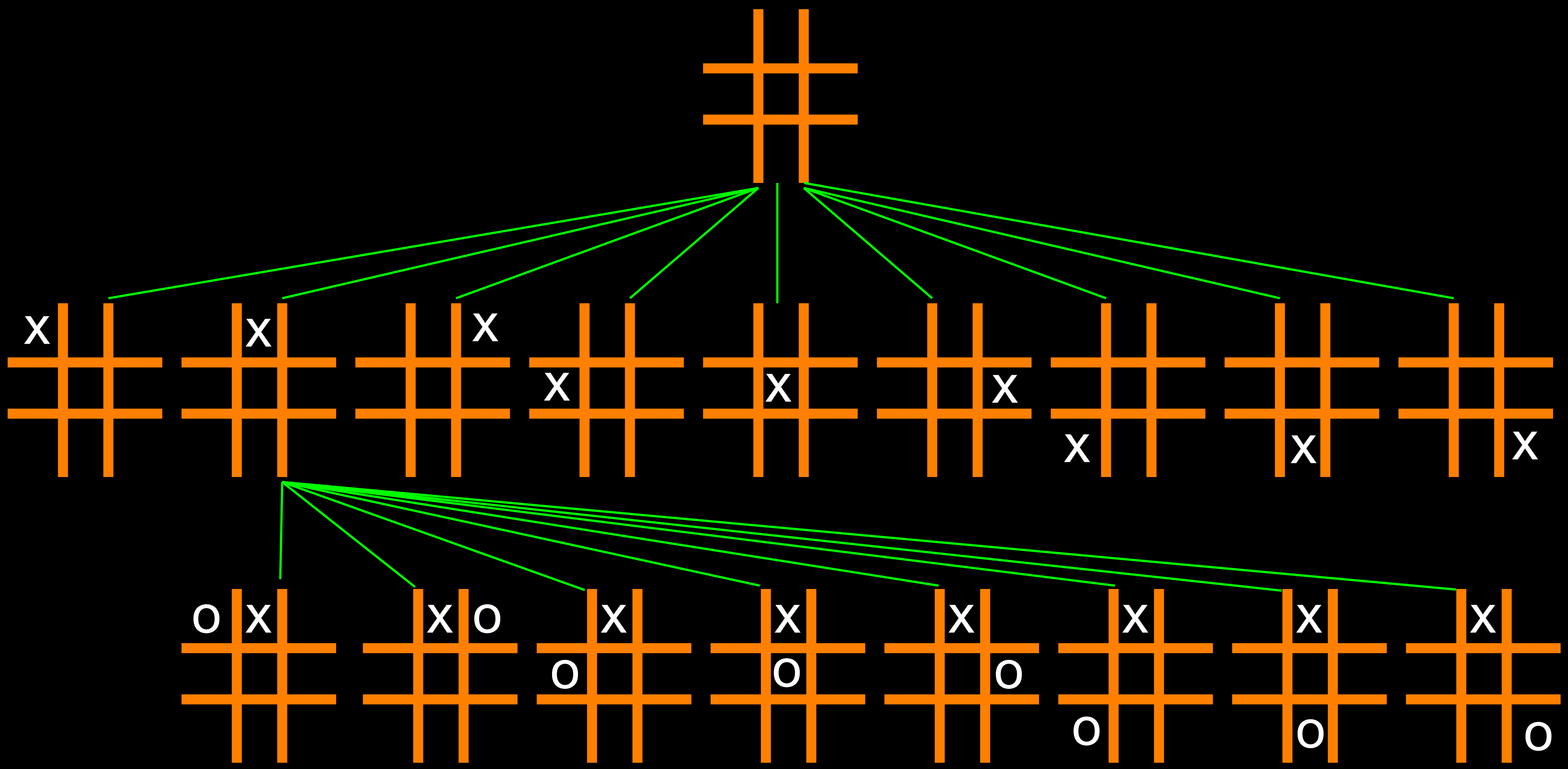
The mini-max strategy consists of three components:

- **Move generator** - determine **your possible moves**
- **Board evaluator** - evaluate the **desirability of each move**
- **Mini-max procedure** - determine an efficient way to search through all the possible moves that you can perform

All of these components use or work upon a ***game tree***.

- A game tree stores, and allows the mini-max procedure to manipulate the possible moves that can be made.

Move Generator Example



Note: Many branches are omitted.

The second level would actually contain 9 x 8 = 72 nodes.

Move Generator Tic-Tac-Toe

Creating a complete game tree starting from the empty state board for Tic-Tac-Toe turns out to be more complex than you might first expect:

- The game tree contains approximately **550,000** nodes.
    - Easy for a computer to handle, but not insignificant.

For more complex games, the complete game tree is effectively unmanageable because the number of possible nodes in the game tree is unbelievably large.

- Therefore, we will not want to construct the entire game tree when making a decision, but rather only construct and **search the most "promising" parts of the game tree.**

Heuristics and ***pruning*** are used to only evaluate the most likely beneficial moves.

Board Evaluator

The second component of a game system is the ***board evaluator*** which is responsible for determining if a given board position or state is advantageous for the player.

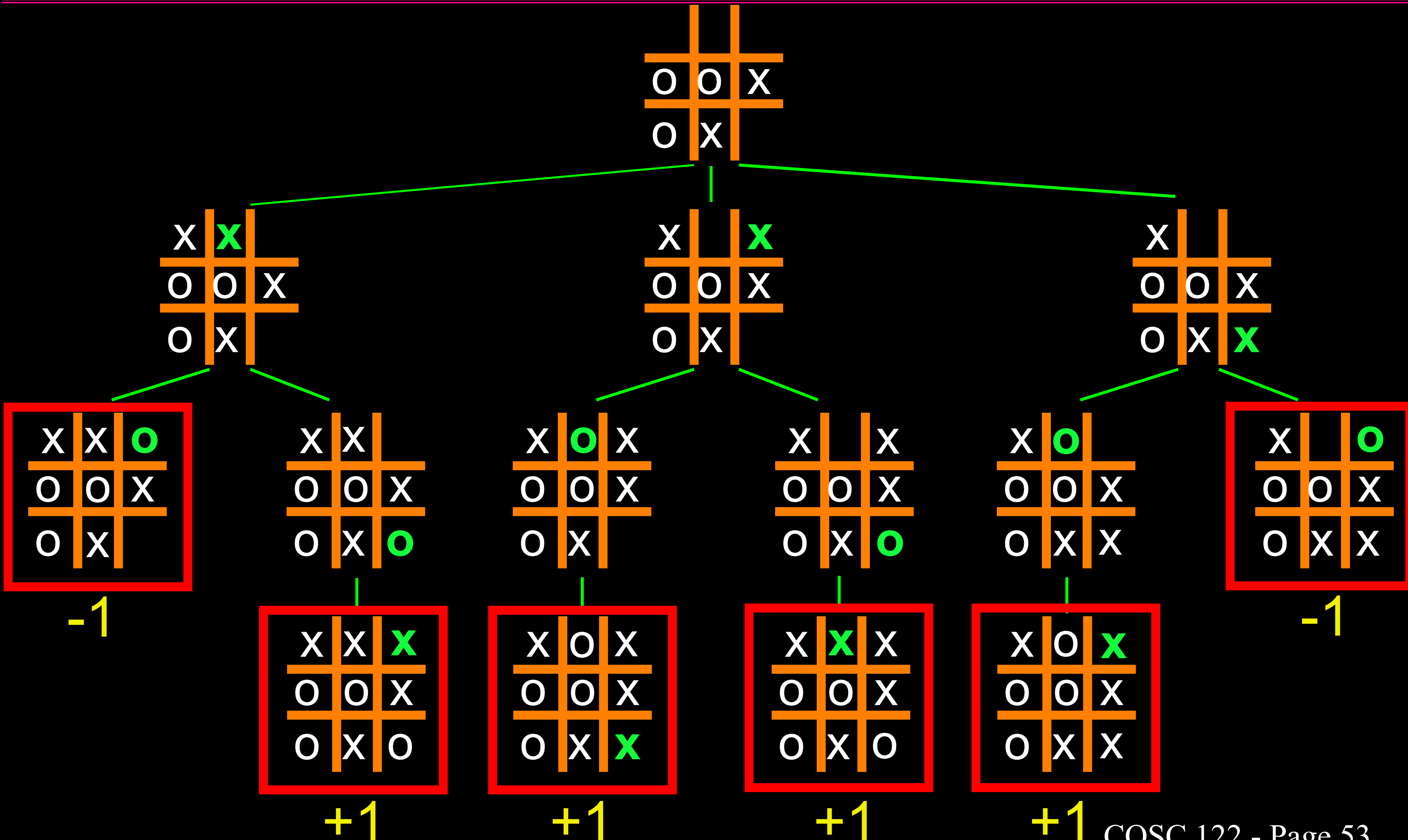
- The board evaluator **determines the good and bad moves**.

The move generator builds a game tree to get some insight as to **what might happen in future moves**:

- Future board scenarios are thus known by playing out moves, counter-moves, counter-counter-moves, etc.
- Future board scenarios are of no use if you have no mechanism to evaluate them.

The board evaluator determines when a sequence of moves (a path in the game tree) is advantageous for the player.

Board Evaluator
Tic-Tac-Toe Example



Why did Deep Blue Win?

Deep Blue ended up winning due to increasing computation power.

- This **extra power** allowed the computer to **examine more possible moves** in the game tree.

The use of ***parallel computers*** that have multiple processors and memory allow for complex problems to be solved.

- The top 500 most powerful computers in the world have **thousands of processors** and are used for simulations of weather, military tests, and earthquakes.

Is Deep Blue intelligent?

- The search algorithm was “intelligent”, but does it qualify as what we consider intelligence?

Computer Creativity

Computers can run programs that **automatically generate music, art, and pictures.**

- The intelligence is still with the software - not the computer.
- The software is **encoding human intelligence**.

The underlying question is: **Is creativity algorithmic?**

- If it is, computers may one day be creative.
- Many things that are creative are algorithmic.
  - Mathematics was once considered creative or inspired.

Creativity is sometimes inspiration but is also a lot of revision.

- Inspiration to create something totally new.
- Revision is modifying existing to produce something new. Algorithmic?
  - Many "new" advertising, research, etc. are based on revisions.

Computer Generated Art

These pictures are generated from algorithms.

Source: Ken Musgrave - http://www.kenmusgrave.com

The Universality Principle

In theory, all computers have **the same ability to compute** as they use the same basic functions.

- This is called the ***Universality Principle***.

In practice, differences in computer hardware, software, and operating systems make it **impossible to run all software on all computers and to run it efficiently**.

Examples:

- programs require processing speed that hardware cannot achieve
- operating systems support different features
- processors encode instructions differently in hardware

Six basic instructions: Add, Subtract, Set_to_One, Load, Store, and Branch_On_Zero.

Why are some programs faster than others?

Recall that an ***algorithm*** is a sequence of steps to solve a problem.

The performance of an algorithm when implemented on a computer depends on the **approach used to solve the problem and the actual steps taken.**

- Example: algorithms for data sorting have different efficiencies.
  - http://www.sorting-algorithms.com/
  - http://bl.ocks.org/andrewringler/raw/3809399/

Although faster hardware makes all algorithms faster, algorithms that solve the same problem can be compared in **a hardware-independent way** using ***big-Oh*** notation.

Algorithms
Best and Worst Case

Very few algorithms have the exact same performance every time because **the performance of an algorithm typically depends on the size of the inputs it processes.**

The ***best case*** performance of the algorithm is the most efficient execution of the algorithm on the **"best" data inputs**.

The ***worst case*** performance of the algorithm is the least efficient execution of the algorithm on the **"worst" data inputs**.

The ***average case*** performance of the algorithm is the average efficiency of the algorithm on **the set of all data inputs**.

Best, worst, and average-case analysis typically express **efficiency in terms of the input size of the data**.

- The input size is often a function of ***n***.

Algorithms
Big-Oh Notation

Big-Oh notation is a mechanism for quickly **communicating the efficiency of an algorithm.**

- Big-Oh notation **measures the worst case performance** of the algorithm by ***bounding*** the formula expressing the efficiency.

In big-Oh notation:

- The performance is specified as **a function of *n* which is the size of the problem**.
  - e.g. *n* may be the size of an array, or the number of values to compute
- Only the most significant expression of ***n*** is chosen:
  - e.g. If the method performs $n^3 + n^2 + n$ **steps**, it is $O(n^3)$.
  - **Significance ordering**: 2^n, n^5, n^4, n^3, n^2, $n*log(n)$, n, $log(n)$
- Constants are ignored for big-Oh:
  - e.g. If the method performs $5*n^3 + 4*n^2$ steps, it is $O(n^3)$.

Algorithms
Common Big-Oh Notation Values

There are certain classes of functions with common names:

- constant = $O(1)$
- logarithmic = $O(\log n)$
- linear = $O(n)$
- quadratic = $O(n^2)$
- exponential = $O(2^n)$

These functions are listed in order of fastest to slowest.

- For example, for large values of *n*, an algorithm that is considered $O(n)$ is faster than an algorithm that is $O(2^n)$.
- Big-Oh notation is useful for specifying the growth rate of the algorithm execution time.
  - How much longer does it take the algorithm to run if the input size is doubled?

How Hard Can a Problem Be?

There exists problems that no computer can solve efficiently.

These problems are called ***NP-complete problems*** and are considered ***intractable***.

- The only way to solve the problem is to try all possible solutions to find the best.
- Even the most powerful computers cannot solve large examples of these problems.
- Example problem: Travelling salesman problem - find best route between *n* cities.

How Hard Can a Problem Be? (2)

Even worse, there exist problems that have been proven to be unsolvable regardless of the computer speed.

There exist no algorithms at all for such ***unsolvable problems***.

An example is the Halting Problem that has the simple task of asking if a given program will always stop (halt) or will it run forever.

Computers in the Future

The future of IT is bright. There are many technologies being developed that are migrating from the research labs into use.

- **Software agents** – Can software be your personal butler?
- **Robots** – When we build robots, what would you want it to do?
- **Self-healing and adapting** – Can our systems fix themselves?
- **Wearable computers** – Can we embed computers in clothing and glasses? In our eyes and brains?
- **Language translation** – Can we have the universal translator?
- **Personal Life Databases** – Can we record all of our life information and moments (text, images, sound, video)?
  - What would that look like? Would you want that?
- **Automatic driving cars** – Our cars will do the driving (probably better than us). They already know where they are going…

Computers in the Future (2)

Some challenges:

- Information overload
  - If we can get data from everywhere at any time, do we get too much?
  - Can we trust the data we get?
  - How about our privacy and security?
- Always-on society
  - Our technology has trained us to be always available for communication.
  - Is that good? Are we actually more productive that way? More human?
- Pace of innovation
  - Technology has sped up society and business. Everything changes rapidly. Innovation may not always be good.
- Essence of Humanity
  - If everything is automated and computerized around us, do we lose the essence of being human?
  - Are we ready for the ability to alter human DNA and lifestyles?

Conclusion

Computers do not yet mimic human creativity although they demonstrate "***artificial intelligence***" in many domains.

- One of these domains is game playing where intelligence is provided by game trees and search strategies.

Computer scientists compare algorithms independently of hardware using ***big-Oh notation***.

NP-complete problems are problems where no efficient solution exists. ***Unsolvable problems*** are problems where it is proven no solution at all exists.

Objectives

- Explain the Turing Test in your own words.
- Define: artificial intelligence
- List and briefly explain the three components of game playing using game trees and the mini-max strategy.
- Define: Universality Principle
- Be able to convert a formula in *n* into big-Oh notation.
- Compare and contrast: best case, worst case, average case
- Compare and contrast: NP-complete problem, unsolvable problem