Web Systems Quiz #2 Notes

Week 7:

Programs vs Processes

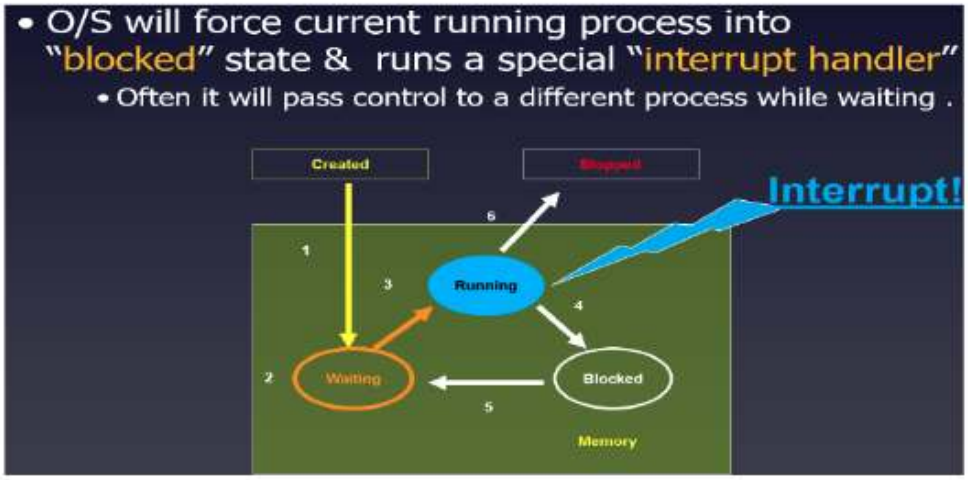
- Processes are programs in execution
- Every time a program is run a new process is created
  - Different processes could be
    - System
    - Services
    - Background tasks

Process Management

- Process management is how the OS handles:
  - Starting processes
  - Managing running processes
  - Performing inter-process communication
  - Terminating processes
- Modern OS has the ability to run multiple processes at the same time
  - Multiprogramming – windows 3 +
  - Multitasking – All modern OS
  - Multithreading-running multiple threads concurrently
  - Multiprocessing

Interrupts:

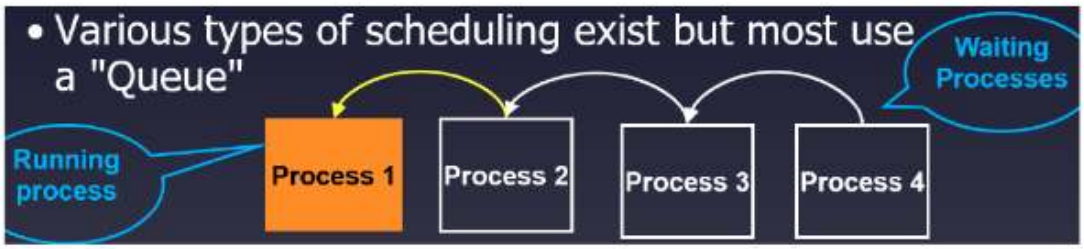
- The OS runs processes until they are interrupted
- Interruptions come from
  - Hardware
    - E.g. ending a task
    - Keystrokes
    - Mouse movements
  - Programs
    - Runtime errors
    - Pausing for input
    - Waiting for load



Process Scheduling

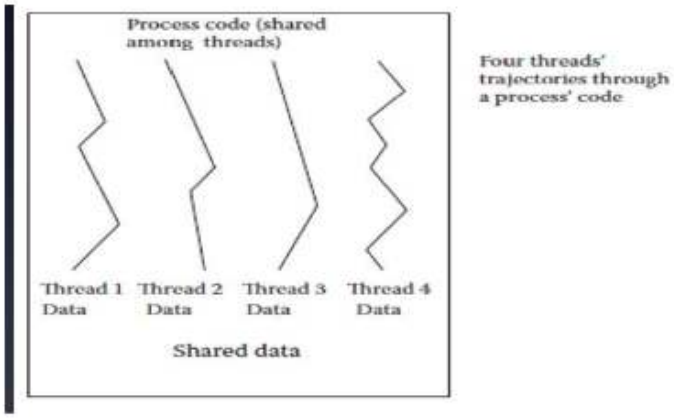
- The OS decides which processes run first through scheduling
- There are many different types of scheduling such as
  - FIFO
  - Pre-emptive
  - Round robin

- We most commonly use a queue



Concurrent Programming

- **Multiprogramming- "cooperative"**
  - When process wait for I/O run next process
  - Can get "hangs" if a process doesn't have I/O and doesn't relinquish the CPU.
- **Multitasking – "pre-emptive"**
  - Set a timer on all processes – "fair share" of CPU time.
  - Has priorities e.g. low, medium, high
  - Choice of queuing algorithm
- **Multithreading**
  - Multiple instances of the same code, same process in memory but 2 or more execution paths
  - Allows an OS to run a process on more than 1 CPI
  - Very efficient use of resources
- **Multiprocessing**
  - Multitasking spread amongst 2 or more CPU's



Inter-process Communication

- Processes often need to communicate to each other this is called IPC
- OS allow communication between processes through
  - File sharing
  - Shared memory
  - Signals
  - Sockets
  - Messages
  - Pipes
  - Semaphores
  - Locks

One-way IPC: Pipes

- One-way communication
- Process A sends data to process B Process B then accepts that data
- Example is a bash pipe operator "|"

Two-way IPC

- Processes can communicate in both directions
  - Shared memory (bit of memory which acts like a file)
  - Named pipe (special 2-way pipe also acts like a file)
  - Socket (either a network or internal interface)
  - Message Queue/ Message passing (special programming interface – passes data like internal messages/SMS)
  - Semaphore (special flag/file which controls access to resources)

Resource Management:

- Resources are the things that processes need to run
  - E.g. files, network, keyboard/mouse input etc.
- The kernel manages all other system resources
  - Interrupts, I/O, system devices etc.
- Many resources require mutually exclusive access
  - i.e. if one process is using the resource no the process can use it until the first process is done
  - results in resource contention

Two-way IPC and Resource contention

- Resource Contention:
  - 2 processes want to alter the same resource at the same time (deadlock)
- Types of resources
  - Memory, files and hardware
- Solutions to this "Deadlock"
  - Semaphores- a flag held by the process changing the memory
  - Lock files- a file is not readable/writable while data is being written into it

- Deadlock is when processes compete for a limited amount of resources
- Sometimes a process has to waif for some resource which is held by another process
- If that resource will never get released for one or another reason this situation is called a deadlock.
- E.g. if 2 users try to edit a file at the same time

What causes a Deadlock?

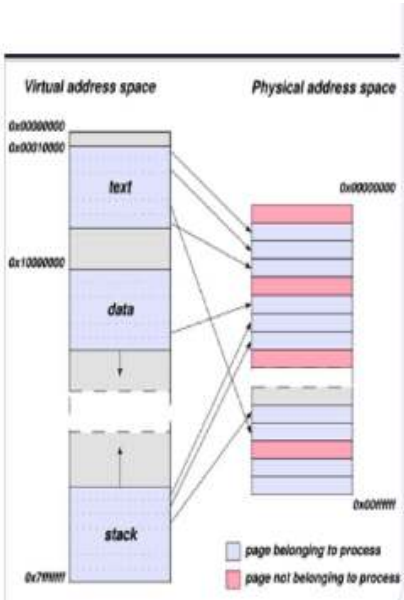
- Mutual exclusion: there exists a resource that can be accessed by only one process at a time.
- Hold and wait: there exists a process that holds at least one resource an is waiting for another resource.

How to deal with deadlocks?

- Avoidance/prevention: OS decides which processes may use resources and when
- Detection/ Management: allows deadlocks to form then finds and breaks them

Memory Management:

- Memory management allows the OS to:
  - Run more processes than we can fit into physical memory
  - Optimise use of expensive Ram
  - Keep track of processes owning blocks of memory
  - Provide access control to memory
  - Decide where process is loaded into memory
  - Handle allocation/deallocation of memory

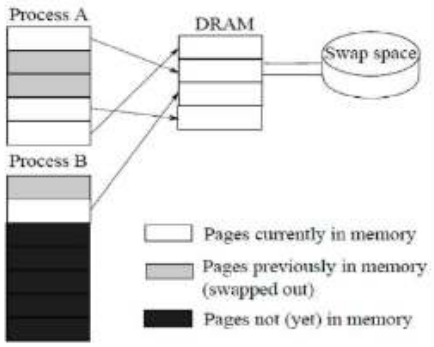


Physical and logical addresses

- All computers have a fixed amount of physical memory e.g. 16gb ram
- Operating systems kernel can directly access physical memory using something called physical addressing
- OS hides physical memory from processes
    - As far as each process is concerned it sees logical memory
- Typical OS will allocate pages (4kb blocks of ram) to a process
- OS translates logical addresses to physical addresses
    - Called logical/virtual addressing
- During execution of a process, the dame logical address may be mapped to many different physical addresses as data and programs are paged out and paged into other locations.
- The logical address space is larger than the physical address space (Ram) if we have virtual memory available

Virtual Memory

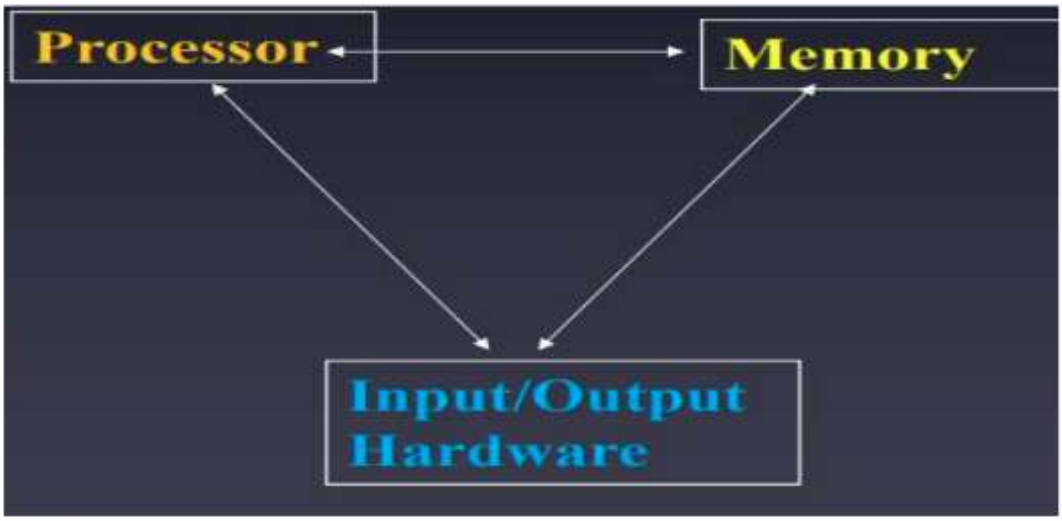
- Virtual memory is a concept which is related to but distinct from the memory hierarchy
    - Don't confuse virtual addressing with virtual memory
- Virtual memory makes part of the hard disk like the main memory to the process
    - This means that programs can appear to have ma lot more memory than is available
    - Virtual memory is much slower than ram
- Terms to memorise:
    - VM: The logical address space
    - Swap file: Part of the HDD used for VM
    - A page: The amount of data which can swap between ram and disk
    - Page table: Maps the logical addresses to either physical or virtual memory
    - Paging: The action of swapping between disk and RAM
    - Page fault: When data to be accessed is not in ram and needs to be swapped from disk
    - Thrashing: When OS spends more time paging than running applications



Locality:

- Paging is slow- minimize to avoid thrashing
- Locality is guesswork on what needs to be in the RAM
- Temporal Locality: recently accessed memory is likely to be accessed again
- Spatial Locality: Locations near recently accessed memory are likely to be accessed soon.

Computer Architecture:



What is a computer?

Memory: Memory stores programs and data, encoded as binary numbers

- Memory types:
  - ROM: Read only memory
    - Permanent information
  - RAM: Random access memory
    - Computer forgets the information when its off
    - Programs change the contents

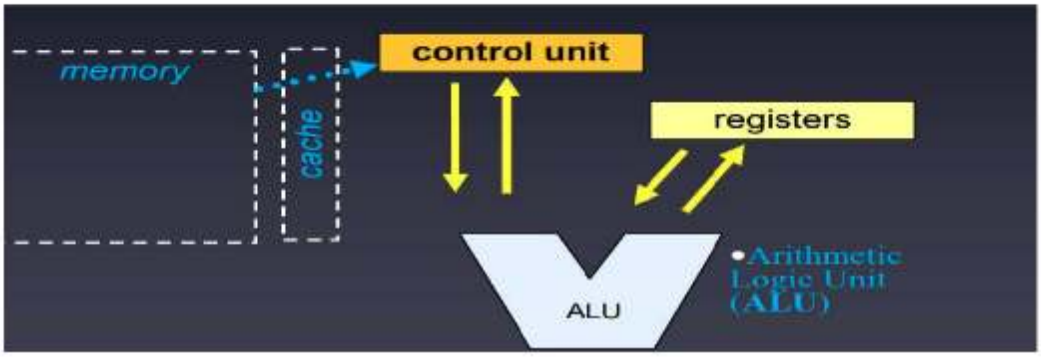
I/O Hardware

- Input devices:
  - Keyboard
  - Mouse
  - Mic
- Output devices:
  - Monitor
  - Printer
  - Projector
- Network devices:
  - Ethernet
  - WiFi
- Storage
  - Disks
  - Tapes

Central processing unit (CPU)

Consists of three main parts:

- Arithmetic Logic Unit (ALU)
    - Performs arithmetic and logical operations
- Control unit
    - Sets up ALU with instructions and data from memory
    - Often uses cache for faster memory access
- Registers
    - Small, fast memory in CPU
    - Loaded by the control unit



Data representation:

All file types on a computer are converted to binary because that's what the computer understands.

Because everything is read as binary until computer O/S designers developed character encoding so that the computer knew how to "read" characters.

Character encoding:

Prehistoric:

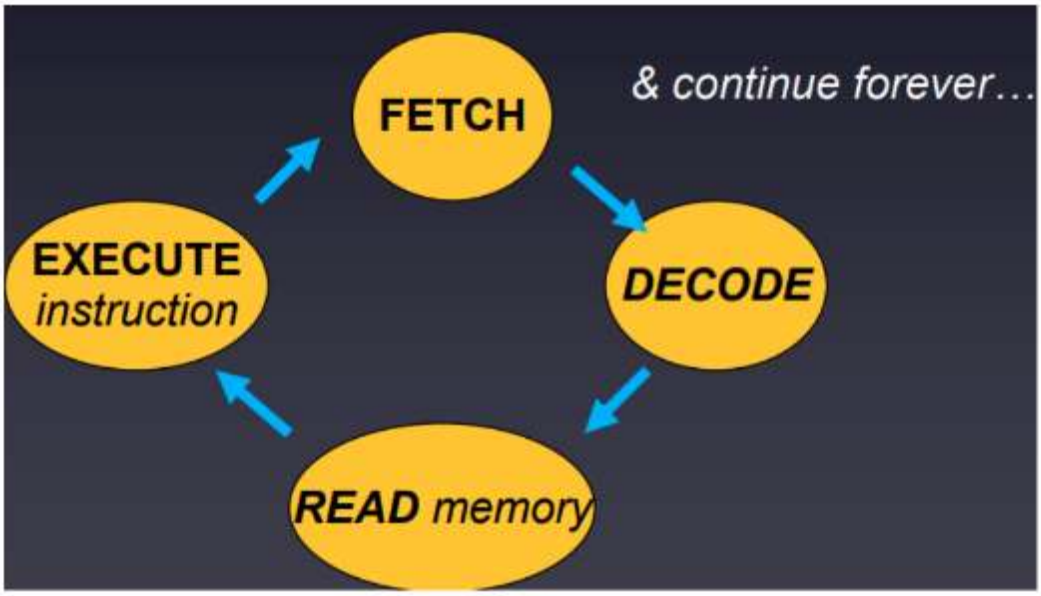
- EBCDIC – Used in IBM mainframes, punch cards

Historic:

- ASCII – 7 bits – 128 different chars
- ISO-8859 – Extension of ASCII to include Latin alphabets

Processor function:

The processor executes instructions on data using a fetch-execute cycle



All CPU's use the fetch-execute cycle

Examples of ACSII:

Bin	01001000	01100101	01101100	01101100	01101111
Dec	104	101	108	108	111
Char	H	e	l	l	o

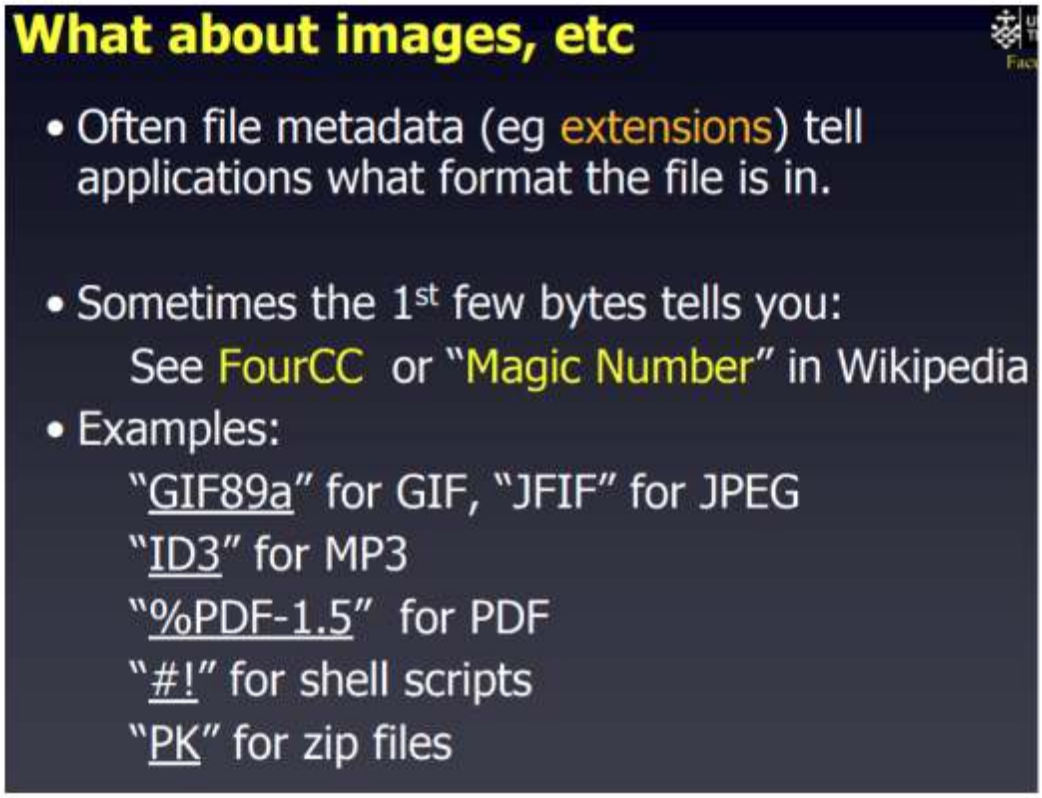
Unicode:

Because there are so many different characters in different languages we cant only use the 128 characters in ASCII so we developed something called Unicode.

- Unicode- Expands ASCII to 16bits or more to represent over 120,000 characters
- But Unicode would take up too much storage so they created UTF-8
- UTF-8
    - Variable length encoding:
        - Character from #00-127 (is one byte)
        - Character > 128 (2 or more bytes)
- UTF-16
    - Every character as 16bits or 32 bits

Example of storage requirements:

Image Storage requirements:



Numbers in conventional computers:

- Data travels along an electrical conductor – typically wires, but now can be optical (or other media)
  - As 'pulse' of voltage – ie 0 volts or > 0 volts (eg 5V) at any given time
  - Assume 0 volts represents 0
  - Assume > 0 volts (eg 5V) represents 1
- A string of 1's and 0's is a binary number

- Note: Many circuits now use 3.6V, 1.8V or lower – needed because of increasing clock speed of chips & energy efficiency.

- A numeral system is a way of representing information, important numeral systems implements on computer include binary, hex, decimal and octal
- The most common numbering system is the Hindu-Arabic system
  - The base dictated the digits available
- E.g.
  - Decimal is 10
  - Binary is 2
  - In computing we use base; 2,8,10,16
- In base 10 we have digits 0-9
  - In base 2 we have digits 0-1
- We use a subscript to depict the base e.g.

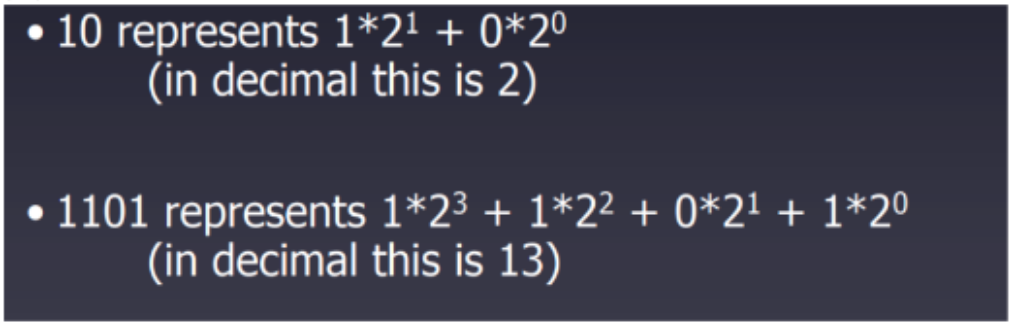
- Eg: Octal: 1234_8
- Eg: Decimal: 314159_{10}

Base 10:

- Uses 10 symbols: 0,1,2,3,4,5,6,7,8,9
- Each symbol represents a quantity
- The decimal system is positional
  - We can represent quantities greater than 9 by putting numbers together

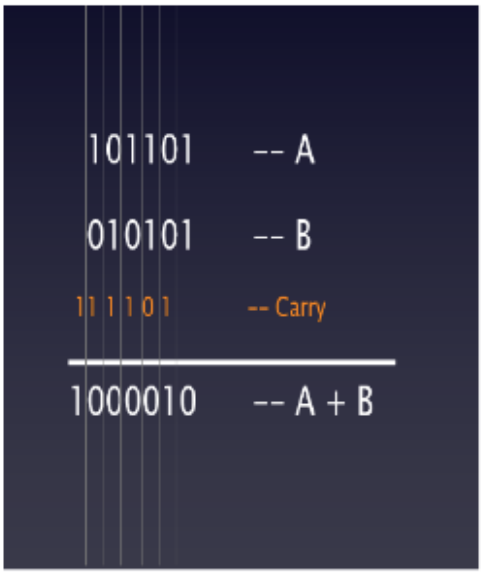
Base 2 - Binary:

- The binary system is positional like the decimal system, but it only has 2 numbers 0 and 1



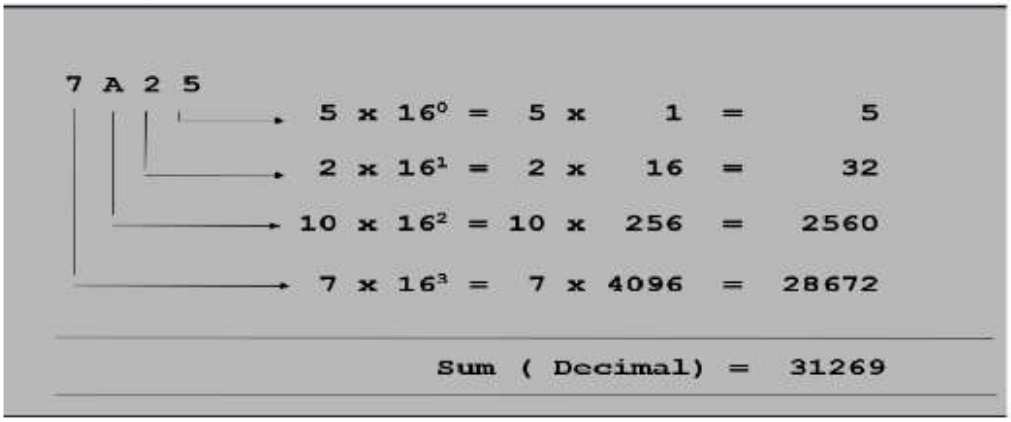
Number Conversions:

Adding Binary Numbers:



Converting Hex to Decimal;

- We multiply each digit in our hex number by the appropriate power of 16
- Example below



Converting Hex to Decimal:

- Repeated division by 16 or
- Repeated subtraction or
- Convert to binary then to hexadecimal

Multiple of 16	Number	Divided/16	Integer Result	Remainder	Remainder (hex)
2	300	300/16 = 18.75	18	.75 * 16 = **12**	C
1	18	18/16 = 1.125	1	.125 * 16 = **2**	2
0	1	1/16 = 0.0625	0	.0625 * 16 = **1**	1

Helpful Notes:

Decimal	Binary	Hexadecimal
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Decimal	2 power	Binary	Hex
1	$_2$0	1	1
2	$_2$1	10	2
4	$_2$2	100	4
8	3	1000	8
16	4	10000	10
32	5	100000	20
64	6	1000000	40
128	7	10000000	80
256	8	100000000	100
512	9	1000000000	200
1024	10	10000000000	400
2048	11	100000000000	800
4096	12	1000000000000	1000
8192	13	10000000000000	2000
16384	14	100000000000000	4000
32768	15	1000000000000000	8000
65536	16	10000000000000000	10000

Power of 16	Hex digit	Decimal
16^0	1	1
16^1	10	16
16^2	100	256
16^3	1000	4096
16^4	10000	65536

- (This is a lecture) OR (This year is 1942) ✓ True
- (This is a lecture) AND (This year is 1942) ✗ False
- (This year is 1942) ✗ False
- NOT(This year is 1942) ✓ True

Boolean Operations and their truth tables

- Every Boolean operation has inputs and outputs
- NOT operation has 1 input and 1 output
- AND and OR operations have 2 or more inputs and 1 output

Truth table for AND:

Inputs: A	B	Output: A and B
0	0	0
1	0	0
0	1	0
1	1	1

Truth table for OR:

Inputs: A	B	Output: A or B
False	False	False
True	False	True
False	True	True
True	True	True

Truth table for NOT:

Inputs: A	Output: not A
False	True
True	False

Truth table for XOR:

Inputs: A	B	Output: A xor B
False	False	False
True	False	True
False	True	True
True	True	False

- eXclusive OR
- More like the "English OR statement"
  - e.g. "Either red or black"
- Equal to the operation
  - (A OR B) AND NOT(A AND B)

Law	And form	Or form
Identity Law	$A1 = A$ The AND operation is only true when both inputs are true. When a Boolean value and a value of 1 (or true) is an input into an and operation, then the value of the output is the value of the other input.	$A + 0 = A$ The OR operation is true when one or both inputs are true. When a Boolean value of zero (or false) and a Boolean variable are the inputs into an OR operation, then the output will be determined by the value of the Boolean variable.
Null (or Dominance) Law	$0A = 0\ (false)$ The AND operation is only true when both inputs are true. When a Boolean value of 0 (or false) is one of the inputs then the output is always false.	$A + 1 = 1\ (true)$ The OR operation is true when one or both inputs are true. When a Boolean value of one (true) and a Boolean variable are the inputs into an OR operation, then at least one of the inputs is true, so the output must be true. The actual value of the variable is irrelevant.
Commutative Law	AB = BA The order of the operation is not important – A AND B is identical to B AND A	A + B = B + A The order of the operation is not important – A OR B is identical to B OR A
Associative Law	(A B)C = A(BC) If we have three or more inputs to an AND operation we can AND any two of them and then AND the result of that operation with the remaining inputs.	(A + B) + C = A + (B + C) If we have three or more inputs to an OR operation we can OR any two of them and then OR the result of that operation with the remaining inputs.
Distributive Law	A (B + C) = AB + AC A AND B OR C is the same as A AND B OR A AND C	A + BC = (A + B)(A+C) A OR B AND C is the same as A OR B AND A OR C

Law	And form	Or form
Inverse law	$A\overline{A} = 0\ (false)$ Note: this expression is the same as $A.\overline{A}$ In many cases the . is left out. The AND operation is only true when both inputs are true. When a Boolean value and its inverse are both inputs into an AND operation, then one of the inputs must be false. Therefore the output is always false.	$A + \overline{A} = 1\ (true)$ The OR operation is true when one or both inputs are true. When a Boolean value and its inverse are both inputs into an OR operation, then one of the inputs must be true. Therefore the output is always true.
Idempotent Law. This is similar to the Inverse Law except that a Boolean variable is both inputs into an AND or OR operation. Although the number of inputs is two, effectively the number of inputs is cut down to one since the two inputs will be identical.	$AA = A$ The AND operation is only 1 (true) when both inputs are 1 (true). When a Boolean variable is made to be both inputs into an AND operation, if the variable is 1 (true) then the output is 1 (true). If the variable is 0 (false) then the output is 0 (false). In other words the output value is the same as the input value.	$A + A = A$ The OR operation is 1 (true) when at least one inputs is 1 (true). When a Boolean variable is made to be both inputs into an OR operation, if the variable is 1 (true) then the output is 1 (true). If the variable's value is 0 (false) then the output is 0 (false). In other words the output value is the same as the input value

Law	And form	Or form
Absorption Law	$A(A + B) = A$ If we expand this expression we get the expression AA + AB. It can be seen that if A is true then AA is true. If AA is true, then the complete expression is true.	$A + AB = A$ If we factorise this expression we get the expression A (1 + B). This means we have two inputs to an AND operation. It can be seen that one input, the expression (1 + B) is always true whatever the value of B. Therefore, the value of the expression becomes whatever the value of A is.
De Morgan's Law	$\bar{A}.\bar{B} = \overline{(A + B)}$ NOT A AND NOT B is the same as NOT (A OR B)	$\bar{A} + \bar{B} = \overline{A.B}$ NOT A OR NOT B is the same as NOT (A AND B)
Double Complement Law	$\bar{\bar{A}} = A$ This law simply says that the inverse of the inverse of something is the thing itself.	

Week 10:

Networks:

- Typical home network called a LAN (Local Area Network).
- **Network:** Collection of computers and devices connected together to allow sharing of resources between users.
- **Network File System:** Part of a remote disk can be made to appear as another hard drive on local PC. Physical location of disk storage is transparent to users.

Types of devices on a network

- **Printer Sharing:** One printer used by several PCs.
- **Device Types on the Network:**
- **Host:** Another name for computer.
- **Modem:** Converts between analog signals and digital signals in dialup access.
- **NIC (Network Interface Card):** Direct connection to LAN.
- **Hub, Switch, Wireless Access Point, Router.**
- **Hub:** Can connect more than 2 hosts. Strengthens the signal. Not concerned with meaning of data. Broadcasts message to all of its ports.
- **Switch:** More intelligent than a hub – looks at **MAC addresses** (physical address of the NIC) in the messages. Provides direct physical connection between hosts when they want to communicate.
- **Wireless Access Point:** Operates on the 2.4 or 5GHz bandwidth (several versions – a, b, g, n). Becoming increasingly popular.

All of these devices together on one network can be categorised as a LAN.

- **Local Area Network (LAN):** Network that connects computers and devices in a geographically limited area. i.e. school computer laboratory, office buildings, etc.

Network Topology:

Refers to structure of the network. There are two types of topologies:

- **Physical Topology:** How the data is actually transmitted. Includes bus, ring, and star topologies.
- **Logical Topology:** Only concerned about where data ends up. Dictates how the hosts share access to the network, avoiding collisions (problem where two hosts transmit at the same time). Includes bus and ring topologies.
    - Physical and logical topology are often different. A LAN with bus physical topology can have a sequential (ring) logical topology.

Types of physical topologies are as follows:

- **Bus (Physical):** All computers directly connect to a common communication medium

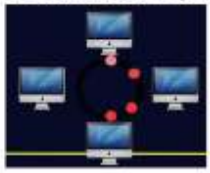
(backbone).
- **Ring (Physical):** All the computers in the network form a closed ring or loop.
- **Star (Physical):** All the devices in a network connect to a central computer or hub.

Types of logical topologies are as follows:

- **Token Ring (Logical):** To avoid collision, hosts take turns to transmit data.

Permission to transmit is called a token. Passed from one host to another according

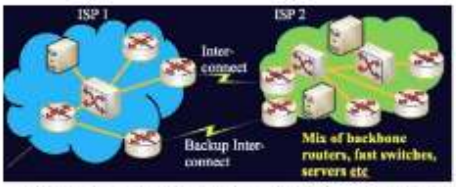
to a set of rules.
- Often, connected physically in a bus or star topology, but computers see this logical



topology as a ring.
- **Ethernet (Bus, Logical):** All hosts have permission to transmit all the time. When collision occurs, wait a random amount of time and try again. If a collision occurs again, double the wait and try again. Network often has a star physical topology.

Internet infrastructure

- **Typical ISP Network:** Called a Wide Area Network (WAN) or Metropolitan Area Network (MAN). Mix of backbone routers, fast switches and servers.



- **WAN:** Network that covers a large geographic area (i.e. a country or the world). Uses **routers** to inter-connect. Uses **modems** to connect to service provider.
- Uses leased lines, circuit switching, packet switching, wireless, broadband networks.
- **WAN connectivity** typically used by ISP companies. Includes leased lines (i.e. HDSL), packet switching (i.e. frame relay), wireless (i.e. 4G, Wi-Fi), Broadband (i.e. ADSL, cable), optical (i.e. NBN).
- **Router:** More intelligent than other devices because it looks at the IP address in the messages. Router decides on the next destination of a message (a packet).
- Simple router is a home ADSL router (technically router + modem). Larger routers found in ISPs.
- Routers use **routing rules** to decide how to transmit messages. Various **routing algorithms** and standards exist (i.e. RIP, BGP).
- Unix command **traceroute** will trace the route to the destination host (i.e. traceroute www.google.com). On windows use the tracert.exe command.

Web Servers: Client-Server:

- Many clients make use of a server. Server provides some shared resource (i.e. processing, large disk, database). Two-Tiers:
- **Fat Client:** Where most of the processing is done at the client side. Just the data is on the server (i.e. Microsoft Access -> SQL server).
- **Thin Client:** Where just the presentation is done at the client and the processing is at the server side (i.e. browser -> web server).
- **Three-Tier Architecture:** Sometimes add a middle tier (more scalable, more security, often used for dynamic web sites).
- i.e. For a shopping website the 1st tier is client (HTML/JavaScript), 2nd tier is web server running PHP, 3rd tier is database server storing orders, inventory.
- Not just retrieval of static HTML pages. Typically, the HTML page is generated dynamically when the HTTP request comes. Dynamic generation is done in the processing tier (or business logic). Generally some of the content of dynamically generated pages is drawn from a database.
- 2nd and 3rd tiers may or may not be on different computers. Three tier applications don't always involve websites.
- **Advantages:** Modularization, can make different user interfaces without altering the underlying processing.

Peer-to-Peer Architecture:

- Alternative to client-server. Clients can also be servers (i.e. Skype, Bittorrent).
- **P2P Distributed Processing:** Idle computers on the internet are a powerful processing resource. Emerging move to use this resource for carrying out many important computations.
    - A processing task is broken up into many parts and distributed to participating computers (i.e. Google compute, Community Grid).
- **Web Applications:**
- **Online Media:** Content continually changing, monitoring visitors, targeting

advertisements according to pages visited, generic browser interaction (i.e. HTTP + HTML/CSS).

- **File Sharing:** File Transfer Protocol (FTP) started in 1971 with Unix. Napster, Bittorrent allowed music and video to be transferred between PCs. Complex technology (not just HTTP).
- **Peer-to-Peer:** Recent surge in interest due to some popular file swapping apps. Three common categories; file swapping (i.e. Bittorrent), instant messaging (i.e. MSN, Messenger), distributed processing (i.e. SETI@Home).
- **P2P File Swapping:** Some protocols published, some not. Incompatible protocols (separate file swapping communities).
- Gnutella protocol published. Leads to many independently written peer programs that are interoperable (i.e. LimeWire, ToadNode). Solves privacy issues with Gnutella and Napster.
- **P2P Searching for Files:** Sometimes search is done by contacting central search server (i.e. Bittorrent tracker portals).
- Search is sometimes **distributed**. Search request forwarded to neighbouring peers. Peers check if they have requested the file. If not, these neighbours forward request to their neighbours. Request stops via a "time to live" field.
- Once the file is found there is direct P2P communication to retrieve it.

Communications:

Oldest is Email. Established in 1972. In 2010, there were 294 billion emails a day (90% spam). Email takes up 28% of workers time.

- Newest is **VOIP**. Computer-to-computer, or computer-to-telephone. Allows voice, video, text messages. Skype is an example of this.(peer-peer)
- **P2P Instant Messaging:** Protocols are typically text protocols (i.e. HTTP, POP3). Different vendors (i.e. Yahoo, Microsoft) use different messaging protocols. Mostly not interoperable.
- Users of different products can't talk to each other. Some communication is possible. Standards work is happening (i.e. SIP, XMPP).

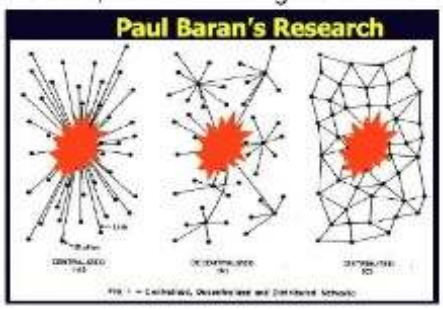
- **Search Engines:** Google's computer visits web sites and follows their links, "Web crawling".

- Creates an index matching combination of words with the URL (HTTP addresses).
- For each search, it ranks web sites according to other website's links to them.
- Earns revenue displaying advertisements aligned with the user's search.

- **Computers and Communication:** Originally computer was just a sophisticated calculator.
- When Unix was developed, using computers to communicate was acknowledged as a necessary function (i.e. by Email).
- Effectiveness of computers is a result of the internet's architecture.

Defence Department & ARPA:

- In 1957 US Defence Department thinks academics in USSR are more successful than academics in the USA. Paid ARPA (Advanced Research Projects Agency) to fix the problem.
- ARPA decides to use computers for communication instead of just for computation.
- **RAND (Research and Development) Corporation:** Paul Baran decides that a network will only survive an atomic bomb if:
- Data between computers has many paths (routing).
- Data is cut up in little blocks, each of which goes on its own journey (packet



switching).

- **Internet Routing:** Data going from one IP address to another. Router forwards packets from one network to another. Packets of the same file can take different paths.
- **Small World or Six Degrees of Separation:** Harvard psychologist Stanley Milgram shows that any two people are at most 6 friends apart.
- **Relationship Graph:** Assume we only directly know friends and indirectly friends-of-friend and so on. Relationship graph should look random.
- **Clustering:** You just know your friends and their friends. If everyone on Earth has 100 friends, the distance between two people on Earth would be 68 million.
- In the real world some people have a few friends far away. Some people have a lot of friends. This randomness can bring the distance from 60 million down to 6.
- **Internet is a Small World:** Nodes are hosts, hubs and switches, lines are Ethernet, ADSL, cable. We communicate using a networking protocol, i.e. TCP/IP.

Unix Existing Network Settings:

- Use **ifconfig** (interface configuration). May need to supply full path **/sbin/ifconfig**.
- i.e. **ifconfig eth0** gives information on the first ethernet interface only.
- Use **ipconfig** for windows.
- **Typical Internet Applications:** Telnet, SSH, web, email, file transfer. These applications use IP addresses and TCP ports for their specific protocol.
- **HTTP (port 80), Network time (port 13), Echo (port 7).**
- **Internet Protocols:** Every client/server application is a set of commands and responses (i.e. HTTP).

Uniform Resource Locator (URL):

Allows identification of resources on the internet. A URL is made up of:

- Protocol/scheme.
- Userinfo (optional)
- Host
- Port (optional)
- Path (optional)
- Query (optional).

HTTP Protocol:

Browser makes HTTP GET request to the web server for a document. The web server locates that document on its hard disk. It then sends the document back to the browser (client) over the network.

- If we type http://www.uts.edu.au into a browser:
- **Browser:** Find out what address www.uts.edu.au is. Query the internet name servers (i.e. nslookup www.uts.edu.au -> 138.25.16.22).
    - Connect to 138.25.16.22 at the port specified by the protocol (this is 80). Send the command "GET /".
- **Server:** Find file index.html and return it.

- **Telnet:** Allows you to login remotely to other networks on the internet.
- No encryption (i.e. telnet rerun.it.uts.edu.au). Runs on **port 23**.
- You can choose other ports explicitly (i.e. telnet start.it.uts.edu.au 80).
- Use SSH instead if it is available.
- **HTTP:** Text based protocol. HTML is a mark-up (not programming) language. HTTP is how we get HTML to our machine. Web browsers render HTML to produce a web page.
- **wget** and **wget -r** http://www.it.urs.edu.au retrieves a file or an entire site.
- **Lynx** http://www.it.uts.edu.au web browser you can use over telnet/SSH. Some systems may have links as an alternative browser.
- **Talking HTTP:** telnet start.it.uts.edu.au 80. Now you are connected to a web server through port 80. Can issue HTTP commands such as GET /index.html HTTP/1.0.
- **Ports:** /etc/services gives a nice listing (for windows, C:\windows\system32\drivers\etc\services).
- ftp-data (20/TCP), SSH (22/TCP), telnet (23/TCP), HTTP (80/TCP), SMTP (25/TCP), etc.

- **SMTP:** Allows for transfer of email between mail servers. Used to **SEND** mail (not usually used for receiving mail). Runs on **port 25**. **Text only** protocol! Consists of 2 parts:

- **TCP/IP is packet switching network**
- **MTA:** Mail Transfer Agent. Uses SMTP (Sendmail, Exim, etc).(a mta is typically a mail server and mail client while an MUA is a mail client)
- **MUA:** Mail User Agent. Uses POP3, IMAP.

- **POP3**: Post Office Protocol. Can retrieve mail remotely. Mail waits on SMTP server until you retrieve it with POP3. **Port 110.**

- **IMAP**: Leaves your mail on the server. Manipulate email from many clients simultaneously. Email is edited whist actually residing on the server. **Port 143.**

- **Character Encoding:** Many disjointed standards. Intended to allow representation of characters as binary numbers.
- **ASCII:** American Standard Code for Information Interchange. 7 bits, extended ASCII is 8 bits.
- **Unicode:** 16 bits. Not all character sets mapped.
- **Text vs Binary:** Files are often grouped into two categories:
- **Text:** Bits correspond to ASCII characters.
- **Binary:** Bits do not correspond to ASCII characters (.doc, .mp3, etc).
- To read either type of file, need the right filter (**cat** understands ASCII, **xmms** and **winamp** understand MP3 – they actually use a library which does the decoding).

Binary Encoding, MIME: Multi-purpose Internet Mail Extensions.

Allows for sending of **binary data** in emails (remember SMTP is a text-only protocol).

- **gmime-uuencode** allows you to encode binary into ASCII (text). Installed on rerun (but not linuxgym).
- **gmime-uuencode some_file:** Check the difference in size of the original file and the encoded output. Every 3 bytes is cut into 4 lots of 6 bits, each a simple ASCII character in a byte of its own. File has 33% more bytes.

Data Encoding, XML: Extensible Markup Language.

Share data in a structured way so it can be easily parsed without ambiguity. XML is meant for sharing data easily (not intended for displaying documents).

- Format is similar to HTML (only stricter). HTML's lack of strictness can be problematic. XHTML addresses this issue.
- Define your own tags (subject to standard rules).

SSH (Secure Shell):

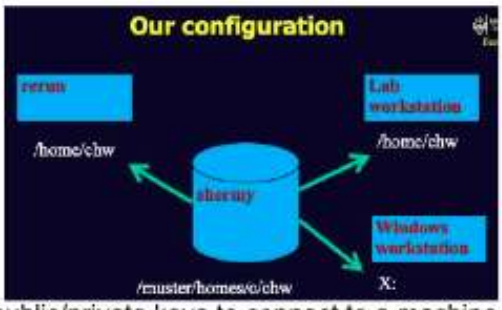
Provides encrypted versions of common network utilities. \
Provides encryption for user pass and net traffic.

- **SSH** is just like encrypted telnet (i.e. ssh rerun.it.uts.edu.au).
- **SCP** securely copies files to remote servers (i.e. scp file1 user@rerun.it.uts.edu.au:/home/user/file1).
- **STFP** is just like FTP only encrypted.

- **FTP (File Transfer Protocol):** Allows you to transfer files to and from different machines.

```
touch testfile.txt
ftp rerun.it.uts.edu.au  (or use sftp for secure version)
cd public_html
get index.html
cd ..
put testfile.txt
help
quit
http://www.uic.edu/depts/accc/network/ftp/vftp.html
```

- **Rerun Files:** Windows uses SMB protocol to share files. SAMBA is Linux's answer to this (reverse engineered by Andrew Tridgell - Australian). Makes a Linux box act like a windows network host.
- **SAMBA** uses ports **137 and 139** (among other ports).
- Supports many other advanced features – some of which even Windows implementation of SMB doesn't support.



- **SSH Keys:** Use public/private keys to connect to a machine with SSH (no need to type password).
- Nearly all machines support SSH2.
- From workstation:

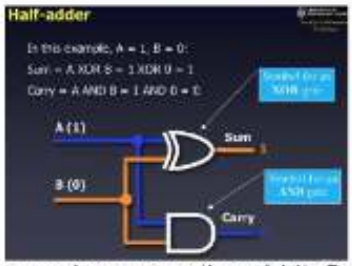
```
ssh-keygen -t dsa
scp ~/.ssh/id_dsa.pub
   rerun.it.uts.edu.au:.ssh/authorized_keys2
ssh rerun.it.uts.edu.au
```

Implement Boolean Logic Using:

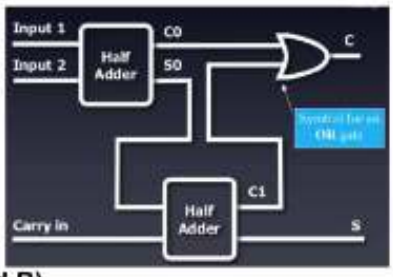
- Electronics (digital circuits), electro-mechanically (usually relays), optics (optical gates), fluidics (water pipes and valves), mechanically (railway switches), even DNA and quantum devices.
- Want to create a logic device which can add numbers.
- **Addition Device:** Produces two outputs. Called a half-adder device.
  - Carry = x AND y.

- Sum = x XOR y.

Input		Output	
x	y	Carry	Sum
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	0



- **Full-Adder:** Required for numbers more than 1 bit. Second and subsequent bits are the sum of **x + y + carry-in**.
  - The circuit to do this consists of **2 half-adders** plus an **OR gate**.
  - To verify we have correctly added two bits (accounting for a carry in), we write the truth table for the "full-adder" functions, with inputs (input1, input2, carry-in) and outputs (S, C).
  - Calculate the truth table for the circuit with columns for (input1, input2, carry-in, C0, S0, C1, C, S).



- **A nand B = not(A and B)**
  - **not(A) = A nand A.**
  - **A or B = (A nand A) nand (B nand B).**
  - **A and B = (A nand B) nand (A nand B).**

4. Even easier conversion between Binary and Octa

Convert the following octals to binary:

a) 127
b) 321

Note : each octal number equals 3 binary digits

Convert the following binary to octals

a) 001 010 111
b) 011 010 001

Q1: Convert decimal 42624 to hexadecimal: **A680.**
Q2: Convert decimal 43176 to binary: **1010100010101000.**
Q3: Convert binary 1000100010011101 to Hexadecimal: **889D.**
Q4: In a multitasking Operating System on a single CPU computer, several processes can be ready for running at the same time but only one of these can actually be running. You can share the CPU workload by: **Putting all ready processes in a queue and allowing them to share the processor and other resources in turn.**
Q5: Convert binary 1000111101010000 to decimal: 36,688.
Q6: Convert Hexadecimal 2557 to binary: 0010010101010111.
Q7: What technique is used by search engines to collect data from the web? Web Crawling.
Q8: What logic operation? X or Y

X	Y	???
0	0	0
0	1	1
1	0	1
1	1	0

Q9: With regards to SMTP, what is the difference between a Mail Transfer Agent (MTA) and a Mail User Agent (MUA)? **An MTA is typically a mail server and a mail client, while a MUA is typically a mail client.**
Q10: Convert Hexadecimal A685 to Decimal: **42,629**

- What is the smallest unit of data to transfer to or from a disk?
  A logical block
- What is the best tag for pre-formatted text?
  Pre
- Which type of security attack would it be if I were to fabricate your login to UTS Online?
  Authenticity attack
- With regards to security, what does confidentiality mean?
  Information is accessible only to the intended recipient.
- How would an index allocation-based file system increase the size of a file when the addresses of its data blocks in the inode are all full?
  Use an indirect block to point to the address of data or more address blocks.
- Which is the most correct statement: in a scripting language such as Bash
  Commands are executed just as if they had been typed on the command line