Class A2 IGCSE Computing

Chapter 32 Number Bases and Representation

Denary Number System

100 10 1

5 8 3

Binary and Hexadecimal Number System

Binary

32 16 8 4 2 1

0 1 0 0 1 1 =19

To convert to Binary, put it under respective heads

(Class HW) Or divide repeatedly by 2s

To convert to Hexadecimal – take the Binary and group it into 4s

0001 0011

1 3 16 * 1 + 1 * 3 = 19

Translate back to Denary – put numbers under the respective heads

Representation of negative numbers using 2s complement

2s complement is used to represent negative signed integers

Consider number 12.

Represent this as 2s complement or -12

Take 12 and convert to Binary (8 bits) – 0000 1100

-12 would be -- 11110100

The one in the beginning denotes it is a negative number.

Rules : 1- Flip the digits and add 1

Or 2 – Starting from right leave all the digits alone upto and including first '1'. Change all the other digits.

Converting -12 to 12 is also similar

1111 0100 -12

0000 1011

\+ 1

0000 1100 12

Or leave all digits alone till first 1. Change the rest.

Converting a 2s complement into Denary

Consider this number : 1001 0111

Put it under the respective heads here and get the resultant Denary number

-128 64 32 16 8 4 2 1

1 0 0 1 0 1 1 1

This equals : -128 + 16 + 4 + 2 + 1 = -105

Binary Subtraction

To convert the number to be subtracted to a negative number, then add it

Consider : 30 – 20

First convert -20 to 20

20 0001 0100

-20 1110 1100

+30 0001 1110

0000 1010 = +10

Chapter 33 Floating Point Numbers

0.5 0.25 0.125 0.0625

.1/2 ¼ 1/8 1/6

Two decimal places in Decimals can hold even 1 paisa (of a rupee)

(discuss – what 2 decimal places in a binary can hold) 0.5,0.25, 0.75

Floating point binary

Consider a decimal – 12000 can be written as 12 * 10^5

Similarly binary – 110.1 can be written as 0.1101 * 2^3

Floating point binary is made up of a sign bit, mantissa (9 bits) and exponent (6 bits).

Example 1 :

0 110100000 000011

0.1101 2^3

110.1 Binary

6.5 Denary

Rules

(1) Place the point between the sign bit and the first digit of the mantissa

(2) Convert the exponent to its equivalent decimal (+ or -)

(3) Move the point right if the exponent is +ve, or left if exponent is –ve, the appropriate number of places.

(4) If the Mantissa is negative and the exponent is negative, move the decimal point to the left adding ones. Else always add zeroes.

(5) Convert the resulting binary number in denary.

Eg 2 :

Now consider the binary split into sign bit, mantissa (9bits) and exponent (6 bits)

0 100000000 111110

This is 0.1 * 2^-2

0.001

That is $1/8^{th}$ or 0.125

Eg 3 :

1 11010000 000011

1.1101 * 2^3

1110.1

If the first digit is a 1 it is a negative number

-8 + 4 + 2 + 0.5 = -1.5

Note : The leftmost bit has a negative place value.

Eg 4 : This is the correct implementation

1 100000000 111110

1.1 * 2^-2 111110 complement 000011 = 2

=1.111

-1 + 0.5 + 0.25 + 0.125 = - .125

Normalization :

Consider 1234000 can be expressed as .01234 * 10^8 or 1.234 * 10^6 or in a normalized form as 0.1234 * 10^7

Similarly for binary numbers – using a mantissa of 9 bits plus a sign bit, the number 0.000001001 would be represented in the mantissa as 0.100100000 with an exponent of 111011 (-5).

Positive numbers : are normalized by getting to the first bit 1 of the mantissa (not counting the sign bit) and adjusting the exponent accordingly.

Eg : 0 000110101 000010

Step 1 : 0.0001101 Exponent 2

Step 2 : 0.110101000 Exponent -1

Answer : 0.110101000 11111

Negative numbers : the normalized form is where the first bit of the mantissa not counting the sign bit is 0.

Eg : 1 111100100 000011

Step 1 : Insert the assumed binary point to the right of the sign bit

1.111100100 Exponent 3

Step 2 : Shift right 4 places, so first digit is a 0

1.00100 Exponent 3-4 = -1 (see here, the 1s are expendable since this is a negative binary).

Answer : 1 001000000 111111

Advantages of Normalized numbers :

Max precision for a given number of bits

Only one representation for a given number.

Chapter 34 Structure and Role of the Processor

The processor consists of 3 main units,

ALU - where Arithmetic & logic operations are carried out.

Control unit – coordinates the activities taking place in the CPU, memory and peripherals by sending control signals to the various devices.

The system clock – which generates a continuous sequence of clock pulses to step the control unit through its operation.

Registers – (32 or 64 bits in size) to hold imformation temporarily as it is being processed.

Program counter (sequence control register, sequence register)– holds the address of the next instruction to be executed.

The general purpose registers are used for performing arithmetic functions. In some computers there is only one general purpose register called an accumulator.

The Current Instruction Register (CIR or the IR) contains both the operand and the operator of the current instruction.

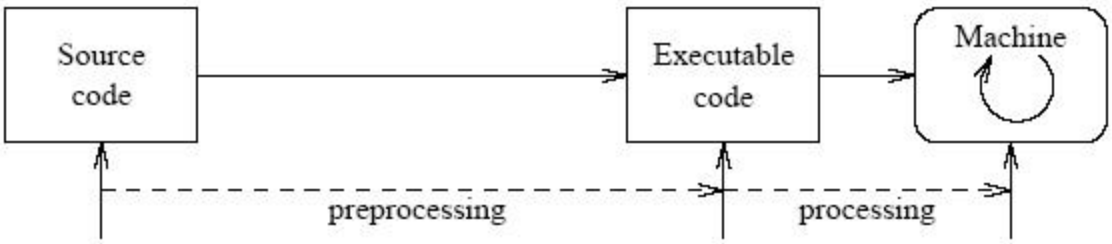
The memory address register (MAR) holds the address of the memory location from which data will be read or to which data will be written.

The memory data register (MDR or MBR) is used to temporarily store data read from or written to memory.

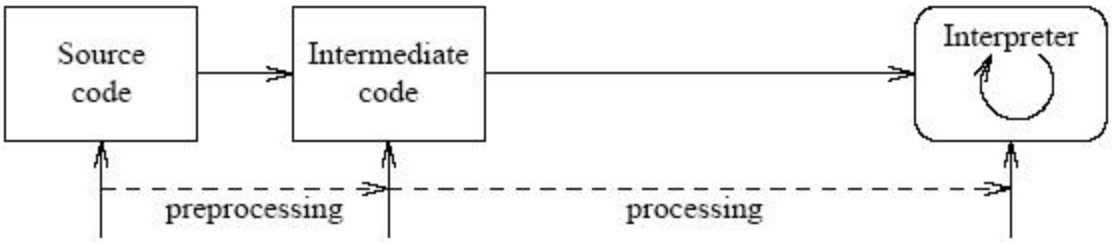
The status register (SR) contains bits that are set or cleared based on the result of an instruction. They also store information about interrupts.

Chapter 37 High Level and Low Level Languages

Low level Languages – these are machine oriented. They are written so as to extract the maximum usage and minimum storage from the computers processing units and memory. A typical example would be a device driver – like a printer driver.



Compilation



Interpretation

1. Characteristic of a low level language are : they are machine oriented and machine dependent.
2. Each assembly language statement (apart from macros) generally translates into one machine code instruction. Hence programming is a lengthy and time consuming exercise.

High Level Languages – are written to solve a particular type of business problem. Pascal was more general purpose, COBOL was for processing files, FORTRAN was more for mathematical problems, PROLOG was for logic and for AI.

Features :

1. In theory they are portable across machines. Java is an excellent example of portability.
2. They are problem oriented.
3. Generally resemble English sentences.

C was developed by Dennis Richie at Bell Labs in the USA inn 1972. It was originally developed for systems programming for the operating system UNIX. It is a relatively low level language which has many of the advantages of assembly languages. At the same time it is easy to learn and can be used like a high level language, it is portable and hides the computers architecture from the user.

Then there is another classification :

Procedural languages

Object Oriented languages

Real time embedded systems : languages for programming real time systems in Nuclear plants, Bomber planes, jet aircrafts, microprocessor controlled washing machines, etc. Time based actions, something needs to happen in 5 seconds, happen every 30 secs and such.

Chapter 38 Object Oriented Programming

Concepts of

Inheritance take a class and inherit from it to obtain its defined properties and methods

Encapsulation hides data, methods

Polymorphism same name method

What are Objects and Classes.

Class – a set of objects that share a common data structure and common behavior.

CarClass

Fields :	Methods :
Steering	Start and Continue Till Stop
Accelerator	

Clutch

Tyres

Objects – Sports Car Methods :

Additional Field Start, Continue, Faster Till Stop

18” Alloy wheels

Containment diagram for a GUI

Event Driven programming.

Benefits of OOP

1. Data is protected and only accessible in well defined ways.
2. Re-use of objects become possible. Specialists designed objects are more reliable.
3. Confusion in the way of multiple functions existing to do similar tasks is avoided.
4. Time Saving
5. Consistency in design and implementation

Chapter 39 Prolog Programming

What is a procedural language. Define ?

Something that follows a sequence, starts of from declaration of variables, to opening up a file, processing the file, doing other things with the data it has read and then closing the file, followed by printing a report of what it found.

What is a Declarative language.

A declarative language does not follow a sequence. It can be written in any order. It consists of Facts, Rules, Goals and Queries.

Fact : A fact consists of a predicate (programmer defined) and zero, one or more arguments.

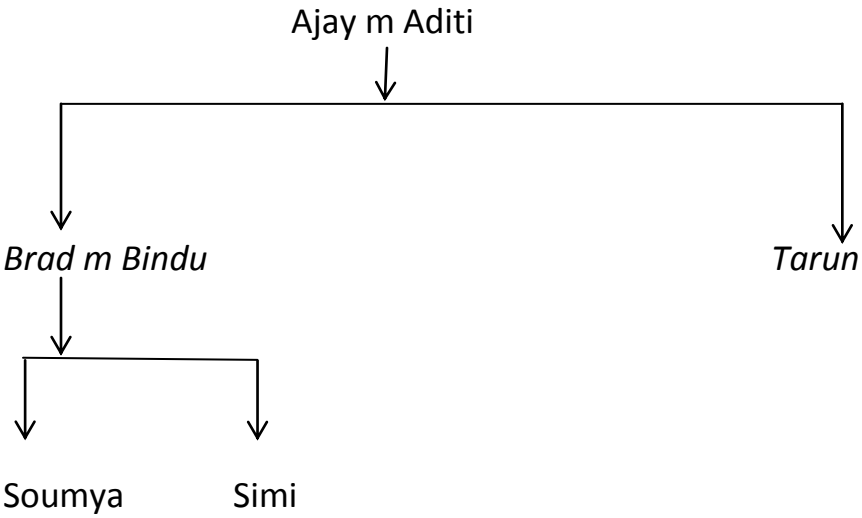
Eg : male(John)

bigger(elephant, donkey)

animal(reptile, large, crocodile)

Atom – start with a lower case letter. They can contain numbers, letters and the underscore.

EXAMPLE :



/* facts */

male(ajay)

male(brad)

male(tarun)

female(aditi)

female(bindu)

female(saumya)

female(simi)

parent(ajay,bindu)

parent(aditi,bindu)

parent(ajay,tarun)

parent(aditi, tarun)

parent(brad,simi)

parent(brad,saumya)

parent(bindu,saumya)

parent(bindu, simi)

/*rules*/

mother(m,x) :- parent(m,x), female(m)

father(f,x) :- parent(f,x), male(f)

grandparent(g,x) :- parent (g,p), parent(p,x)

brother(x,y) :- parent(z,x), parent(z,y), male (x), not (x=y)

sister(x,y) :- parent (z,x), parent(z,y), female(x), not (x=y)

Does simi have a sister

?- sister(x,simi)

X= Saumya

?- mother (x,bindu)

X=aditi

?- brother (who, simi)

no

Entering items in a Prolog database

You can enter items in a Prolog database using the keyword '**assert'**. Assert is also called a **predicate** (built-in) in Prolog's built-in database and succeeds by asserting its argument. [Not and listing are other built in predicates]

assert(carnivore(lion)).

assert(eats(X,meat):-carnivore(X)).

Rules : A rule is an IF condition in Prolog which is used to extend facts.

Consider :

```
bigger(elephant, horse).
bigger(horse, donkey).
bigger(donkey, dog).
bigger(donkey, monkey).
```

If one asks

?- bigger(elephant, monkey)

Prolog will respond with No. Because there is no fact that state this. Still we know it it true because of the transitivity rule. We can define the rule in the program and subsequently it will work.

```
is_bigger(X, Y) :- bigger(X, Y).
is_bigger(X, Y) :- bigger(X, Z), is_bigger(Z, Y).
```

:- IF

, logical AND

; logical OR

= tests whether 2 things are identically the same

__ anonymous variable (eg:used to find all those people who are mothers)

Changing facts at run time (see page 237) for example.

EXAMPLE :

Chapter 40. Recursion

In a normal scenario you would define a function and call the function from another place.

In recursive functions, the function is called from within itself.

Consider the example of a factorial

1. If n =1, then Factorial of n = 1
2. Otherwise, Factorial of n = product of n and Factorial of (n-1)

```
Int Factorial (int n)

{

If (n==1) then return 1;

Else

Return Factorial(n-1) * n;

}
```

Equivalent For loop

```
Result = 1;

N = 10;

For (i=n; i>=1; i--)

{

   Result = Result * n

}
```

Recursive functions are evaluated at the time of unwinding:

Fact(4) = 4 * fact(3) = 3 * facty(2) = 2 * fact(1) = 1

24 <- 4*6 <- 3*2 <- 2*1

If the recursion is for a lot of numbers then unwinding can result in stack overflow.

Advantages : Non-recursive solution is more efficient in terms of computer time and space.This is because when using a recursive solution, the computer has to make multiple function calls, storing return addresses and copies of local and temporary variables, all of which takes time and space.

Also, the values/return addresses are put into a stack during the run and stack overflow will happen if the program runs too long.

In general if a recursive algorithm is much bigger than a non-recursive one then use the non-recursive one.

Chapter 41 Lists

Think arrays.

. u can insert or delete items from any point in a list

. data items may be repeated in a list

. list may contain any type of object

. a particular list may contain different object types.

Colours = [red,green,blue,indigo,violet]
Class = [1,44,0.4,[a,hj,ll],23]

Definition of a list in Prolog

Consider

Colours = [red,green,blue,indigo,violet]
Here the Head is red and the Tail are the others

Colours= [red|green,blue,indigo,violet]

[a|[b]] is a valid list but [a|b] is not since b is an atom and is not a list.

Operations on Lists :

Head[List] returns the element at the head of the list if the list if non-empty, otherwise reports an error.

Tail[List] returns a new list containing all but the first element of the original list.

Empty[List] returns TRUE if the list is empty or FALSE otherwise.

A recursive procedure to print out a list

The procedure keeps on recursively calling the procedure T(List), not executing the Print statement until the list is empty and the procedure runs to completion, when it begins to 'unwind'. It thus prints the list in reverse order.

```
Procedure T(List)
If not Empty(List)
Then
   T(Tail(List));
   Print(Head(List));
End if;
EndProc
```

Elements of a list may be held in an array.

Finding an item in a list

Inserting an item from a list

Deleting an item from a list

1. FINDING : Declare an array and write the pseudocode to find an item which is entered by the user in an ordered list.

   ```
   Procedure Find_Item
   Begin
   ```

variable = item to be searched

p=size

found = false

while item[p] > variable

p = p-1

endwhile

// when p = 0 the item is not in the list

// if p > 0 then item may be in the list

If p <> 0 then

If item[p] = variable

then found = true

// p is the found position, item[p] is the found item

end if

end if

End Procedure

2. INSERTING : Declare an ordered array with say 6 items.

   Put the new item in a variable.

   See if the list is full, if full, display message “List Is full”

   Otherwise, start at the end of the list and examine each item.

   If value of the last item is greater than value of the new item, move current item down one place (to p+1)

   Procedure Insert_Item

```
Begin
    Get new item
     variable = new item
    P = size
If size = max then write "List is full"
Else  //makes space for the new item and finds the correct space
  While item[p] > variable
      item[p+1] = item[p]
      p = p-1
 endwhile
 size = size + 1 //adds in the new item
 item[p+1] = variable
end if
End Procedure
```

3. DELETING an Item from a list

```
Procedure Delete_Item
Begin
    Get item to delete
    Call FindItem to see if item is in list, and to return position  if found
    If found = false then write error message
    Else
            While p < size
```

```
            item[p] = item[p+1]
            p = p + 1
        Endwhile
        size=size-1
    end if
End procedure
```

Chapter 42 – Linked Lists

A Linked list is a data structure used to hold a sequence.

Each item is called a node and it contains a data field and a next address field which is also called a node or a pointer.

A Linked list also has 2 additional pointers

start – pointing to the first element of the list

nextfree – pointing to the next free element of the list.

Algorithm to insert an item into a linked list

Consider this list :

start = 1

nextfree = 5

Address	Name node[p].name	Pointer node[p].pointer
1	B	4
2	T	0

3	J	2
4	C	3
5		6
6		7
7		0

To insert M

Put M in 5

M goes before T -> so M pointer should point to T

M goes after J -> so J pointer should point to M

nextfree should be 6

Begin process

Node[nextfree].name = newname

P = start (now get into a loop)

follow pointers until node[p].pointer points to a name > newname

(this should lead u to J whose pointer points to T which is > newname)

temp = nextfree (which is 5)

nextfree = node[nextfree].pointer (it will set nextfree to 6)

node[temp].pointer = node[p].pointer (M pointer will be set to 2)

node[p].pointer = temp (J pointer will be set to 5)

End process

Handling 2 special cases :

1. If list is full.

   At the start of the algorithm you can add :

   If nextfree = 0 then

      Write “List is full” and exit procedure.

2. If you want to insert into an empty list :

   In an empty list pointers will be populated and start will be 0 and nextfree will be 1.

1	2
2	3
3	4
4	5
5	6
6	0

   If start = 0 then

      Temp = node[nextfree].pointer (node[1].pointer = 2)

      Node[nextfree].pointer = 0 (node[1].pointer will be set to 0)

   (there are 2 zeros in a linked list, one at the position where next item is not there and one at the last element)

      Start = nextfree (1)

      Nextfree = temp (2 from statement 1)

   Else

P = start

End if

Deleting an item from a linked list :

Deleting J :

1. Follow the pointers till J is found (C)
2. Change C pointer to point to T
3. Change J's pointer to point to nextfree
4. Change nextfree to point to J

Pseudocode :

Begin process (delete J)

P = start

Follow pointers until node[p].pointer points to the name to delete

(C is our element that points to J)

temp = node[p].pointer (put 3 in temp)

node[p].pointer = node[temp].pointer (put 2 in Cs pointer field)

node[temp].pointer = nextfree (put 5 in J's pointer field)

nextfree = temp (put 3 in nextfree)

End process

Again here there are 2 special cases :

If start = 0 then write "Nothing to delete" and exit

Start = 1..2

Nextfree = 3..1

1	B	2...3
2	X	0
3		0

If deletename = node[start].name then

temp = node[start].pointer (2)

node[start].pointer = nextfree (3)

nextfree = start (1)

start = temp (2)

End if

Print out a Linked List

Begin process

p = start

While p <> 0

current = node[p] (what is the element I want to print)

print current.name (the print statement)

p = current.pointer

(next value will be the pointer of current value)

End While

End Procedure

Chapter 43 : Stacks

First in Last Out

A stack is also implemented as an array

Top = 3, MaxStackSize = 6

6	
5	
4	
3	Anne
2	Millie
1	Charles

Procedure Push

If Top = MaxStackSize

    Then Write “Stack is full”

Else

    Add 1 to Top

    Stack[Top] := NewItem

End If

EndProc

Procedure Pop

    If Top = 0

    Then Write ‘Stack is empty’

    Else

PoppedItem := Stack[Top]

Subtract 1 from Top

EndIf

EndProc

Use of Stacks :

1. Evaluating expressions like X=(A+B)*(C+D)

   First add A and B and store it in an address.

   Then add C and D and store it (Push) it on another address in a stack.

   Then retrieve both one by one and multiply and Push it again onto stack.

2. Use of a stack for storing the subroutine return address in recursive routines. As each sub is called the machine saves the return address of the calling program on a stack. This ensures that the sub returns are processed in the reverse order of subroutine calls.

Chapter 44. Queues

First in First Out

A queue can be implemented as an array with

Front, Rear, MaxSize, NumberInQueue all populated.

Circular Queue : A circular queue is something where elements do not move to front of queue when an item is removed. The space is vacant. The space is occupied when new elements are added in a circular fashion.

Procedures for Circular Queue

QueueSize

FrontP

RearP

NumberInQueue

To initialize a Queue

```
Proc Initialize
    Front := 1;
    Rear := 6;
    NumberInQueue := 0;
EndProc
```

To add an element to a Queue

```
Proc AddElem
    If NumberInQueue = 6 then
       Print “OverFlow”
   Else
       If Rear = 6 then  // or Rear := (Rear Mod 6) + 1
            Rear = 1
    Else
            Add 1 to Rear
    End If
    Q[Rear] = NewItem
```

```
        Add 1 to NumberInQueue
        End If
    EndProc
```

To remove an element from a Queue

```
    Proc RemElem
        If NumberInQueue = 0 then
                Print “Q empty”
        Else
                NewItem := Q[Front]
                Subtract 1 from NumberInQueue
                If Front = 6 then Front = 1
                Else Add 1 to Front // Front := Front Mod 6 + 1
                End If
        End If
End Proc
```

Use of Queues :

1. Print jobs waiting to be printed.
2. Jobs waiting to be scheduled to run

Chapter 45 : Trees

Chapter 46 : Searching and Sorting

Searching Algorithm

Linear Search :

Start at beginning of list

Repeat

    Test next item for a match

Until item found or end of list reached

Binary Search :

Procedure Binary Search :

```
ItemFound := False;
SearchFailed := False;
Top := N;
Bottom := 1;
Repeat
    Midpoint := Integer part of ((Top + Bottom)/2)
    If A[Midpoint] = ItemSought
            Then Found := True;  // get this away with
    Else
            If Bottom > Top
              Then SearchFailed := True  // get this away with
              Else  // main algo
```

```
                If A[Midpoint] < ItemSought
                        Then Bottom := Midpoint + 1
                Else Top := Midpoint – 1
                End if
        End if
    End if
    Until ItemFound or SearchFailed
End Proc
```

Sorting Algorithm

The Classic Bubblesort

```
Procedure BubbleSort
    Repeat
        Flag := False
        For Count := 1 to N – 1
            If A[Count] > A[Count + 1] then
                Temp := A[Count]
                A[Count] := A[Count + 1]
                A[Count + 1] := Temp
                Flag := True
            End If
        End For
```

```
            Subtract 1 from N
    Until Flag = False or N = 1
EndProc
```

Chapter 47 – Operating System Classification

Operating System : program that controls the execution of all other programs (applications) and acts as an intermediary between users and computer.

What is booting : Operating system is held on a disk and has to be loaded into main memory when a computer is switched on, before any other programs are run. This process is called booting.

The ROM contains the programs that will tell the computer where it can find the part of the operating system which needs to be loaded first into memory. Once this part is loaded, more instructions are executed to load the rest of the Operating system needed for the applications programs to run. This process is called “pulling itself up by its bootstraps” or booting.

Types of Operating Systems :

Batch

In a batch operating system, a job runs from beginning to end without intervention from the user. The running of batch jobs is normally controlled by a JCL – do an example of a batch script with the students.

Interactive

Here the user and the computer interact with each other. User types in commands, supplies data and the Operating system responds with results.

Real Time

This is again classified as – Process control, Information storage, and retrieval and transaction processing.

Processor should not be used to full capacity for most of the time.

These should respond immediately to input. Event based.

These should not have a time lag of even a few seconds.

These should be fault tolerant – error reporting and things like that are of no use here. It has to succeed

Examples :

Nuclear reactor safety system

Airbag deployment system in cars

A real time system should be able to respond to events happening at unpredictable times and which may happen in parallel.

Now for the other Classification :

Single User / Single process

Multi Programming -running more than one program on a machine

Multi Tasking -running more than one task/program on a multi user system. Processor will assign time slices

Multi User (ability to support multiple users as a whole in a networked environment, processor will assign time slices. Multiple users are able to communicate with the computer at one time)

Multi Processing

Multi Threading

Other types of Systems :

Network Operating systems :

Similar to Multi User systems, an OS that lets network connections to happen either as peer-peer or as a Client Server

Client Server systems :

A network organization in which work stations make use of resources available at one or more servers (printer server, file server, database server, application server, web server …)

Distributed systems :

A system in which resources like disk storage and printers exist in separate nodes on a network and users can access these resources transparently, as if they are connected to their own machine.

Chapter 48 - Operating System Concepts

User Interface : command-line (CUI)

Job Control Language (JCL)

Graphical User Inetrface (GUI)

Command Line Interface :

Practice writing some batch files in command prompt. Executing of batch files. Scheduling of batch jobs. Example of a batch job – sending letters to students whose books are overdue.

Job Control Language :

This is a batch program in a typical mainframe computer where jobs are executed offline based on the JCL job that has been written. Priority, Error processing, Time for the run and Output are all part of the JCL.

Graphical User Interface :

The GUI is used to interact with the user in a more user friendly way. GUIs are typically a slower interface. Repetitive tasks are best done using a command line interface than a GUI.

GUI are aimed towards the novice user. Comprehensive online help is also available. There is a consistency of layout as well.

Disadvantages over CUI :

They use more main memory and hard disk space

They require a more powerful processor and a better graphics display

They are slower while executing a command because much more interpretation takes place.

They can be irritating to use for simple tasks because a greater number of operations is required.

Operating System Functions :

1. Process management
2. Memory management
3. I/O Control
4. File management

The Process concept :

A Process is a program in execution.

A process may be in any one of 3 states :

1. A process **is running or current** if it is actually using the CPU
2. A process **is runnable or ready** when it could use the CPU when it is available.
3. A process **is suspended or blocked** when it is waiting for I/O and could not use the CPU even if it were free.

Process Control Block (PCB)

When a process is interrupted it saves the place where it needs to resume next time it is started in a structure called the Process Control Block.

This will contain the following information :

Process_id

Current_process_state

Job_priority or other scheduling information

Program Counter

Register save area

A pointer to the processor's allocated memory area

Pointers to other allocated resources

CPU time used so far

Estimated time to completion

Links to other PCBs

Threads :

A thread is a part of a program that is run asynchronously and the rest of the program continues on its path.

Consider a calculator program that is executing a very complex task and is taking time. Now if you want to provide a feature whereby you want the application to be able to stop when a button is pressed, such a feature is provided by means of threading.

Allocating job priorities :

The scheduler will be responsible for allocating job priorities and to assign a number which will determine what job will run when.

Scheduling objectives –

A scheduling policy should try to

1. Maximize throughput – try to process as many jobs as possible in as little time as possible.
2. Maximize the number of users who can work optimally quickly and achieve good response times.
3. Balance the resource use – if for example a printer is idle a high priority could be given to a job using the printer.
4. Give priority to jobs based on how long they have been waiting as well.
5. Achieve a balance between response time and utilisation of resources.

Criteria used for scheduling :

1. How much I/O a process needs.
2. How much CPU time a process needs.
3. Whether the process is batch of interactive

4. How much more time a process needs to complete, or how big a process is.
5. How important is fast response for this job.

Deadlock of jobs – one resource is waiting on a resource that the other job is using and vice versa.

Chapter 49 : Memory, File and I/O Management

Objectives / Why is memory management needed :

1. To enable different processes to be executed concurrently. Allocate appropriate memory resources.
2. To protect processes from each other when they are running together.
3. To enable memory sharing of memory space when required.
4. To make the addressing of memory space as transparent as possible.

A **Loader** loads the process/program that needs to be executed.

Absolute loader : loads the program into a fixed area in memory. It is fixed at compilte time and the program will work only if the program is loaded at that specific area in memory.

Relocating loader : loads the program anywhere in main memory and all addresses subsequently are relative to the start address. The start address is loaded into a special register called the ***base register.***

There are again 2 types of relocation.

Static Relocation : Once loaded into a position it cannot be relocated again.

Dynamic relocation : Once loaded into a position it can be relocated again. This is necessary for multi-programming OS.

This is made possible by not replacing any logical address references with physical addresses. The logical to physical mapping is done at run time by

the hardware Memory Management Unit (MMU) using base register addressing.

Virtual Memory : memory on hard disk where pages of program and data are stored temporarily.

It makes believe that the computer has more physical memory than it really has.

Main memory is divided into fixed size blocks (typically 4K) called frames. Each process (logical memory) is divided into blocks of the same size, called pages.

Pages reside in virtual memory and are swapped in and out of main memory when required by the process.

An address of a location in a program is of form (p,d)

Where p is number of page, d is displacement from top of page.

Dynamically Linked Libraries (DLL):

Is a pre-compiled and linked executable file. It is linked to the executable and parts of the DLL are loaded when needed by the main program. This saves memory though takes a little more time.

You put shareable code into a DLL. You also put code which is not always needed into a DLL.

File Management :

Functions : (1) To allocate space on the storage device when a file is loaded and to deallocate space when a file is removed.

(2) Files need not occupy contiguous space. They may be divided. OS has to take care of this. When a file is updated and new records added, these may occupy a disparate location.

(3) To keep track of allocation units occupied by each file.

A disk is divided into partitions of logical drives (known as volumes in NTFS – D,E,F,G,etc).

Each partition is divided into clusters (4kbytes) – each composed of sectors.

Each cluster is allocated to a file or a folder or is free. A table called the File Allocation Table (or FAT) details the contents and status of each addressable block on the disk. FAT is also stored on disk.

Input/Output Management :

CPU communicates with I/O devices using an I/O bus. Each I/O device has a hardware controller unit attached to this bus system which can transmit data to or receive data from main memory. Each device sends an interrupt signal to the processor when it has completed its task.

Device Driver:

Breaks down a user request for an I/O device to do something into a series of sub tasks. Like – check if the disk is present and active, locate the space for the file on the disk, position the read write heads at the appropriate write location … all needs to be done to save a file to disk.

Interrupt Handling:

An interrupt is like an event in event driven programming.

When the CPU needs to tell the I/O device something it will have to send an interrupt. When the OS is ready to service the interrupt it will save the state of the interrupted process and passes control to the appropriate routine.

Types of Interrupts :

(a) Interrupts by the running process – say to perform I/O, obtain more storage.

(b) I/O interrupts – are initiated when say an I/O operation is over, an error has been detected or say when a device is made ready.

(c) Timer interrupts – when say a time has lapsed and it is now the turn for another process to run.

(d) Machine check interrupts – malfunction of hardware

(e) Program check interrupts – caused by errors within the program like say reference to an object not in scope.

Example of an interrupt mechanism

Program A is running

It requires to do some disk I/O, so an interrupt is generated.

The interrupt handler changes A to 'blocked' makes a request to the disk drive to get data and invokes a program called the dispatcher which selects Job B to run next.

Sometime later, the disk drive fills up the buffer and says it is ready.

Program A is now changed to 'runnable' by the interrupt handler.

One millisecond later the status of B is changed to 'runnable' and the processor hands over the CPU to A.

Chapter 50 Database Concepts

Diagram of DBMS in an office to get a reference point.

Traditional File Approach

- Data Redundancy
- Data inconsistency

- Program Data independence – adding fields, change in format
- Lack of flexibility – non routine report, assimilate data takes time
- Data was not shareable – across departments

Database approach

Database – collection of non-redundant data shareable between different apps.

Unproductive maintenance – programs were still dependent on the structure of the data so that when one dept needed to add a new field to a file all other programs accessing that file needed to be changed.

Lack of security – All data was accessible by all apps.

Database Management System (DBMS)

- Allows for program data independence whereby the storage/structure of the data is hidden from the user
- Restricted user access to data – provides for security.

3 level architecture

External or user schema – individuals view of DB

Conceptual or logical schema – over all view of db including entities, attributes and relationships

Internal or storage schema – How the data is stored in the db. What sections it uses, etc.

Functions of the DBMS

1. Data storage, retrieval and update.
2. Creation and maintenance of data dictionary.
3. Allow and resolve issues while sharing the db

4. Backup and Recovery
5. Security

Multi Access DB – a DB which allows more than one user to access the DB

Ensuring Integrity of the DB –

Example of 2 users trying to access the DB at the same time. One user could potentially override other users' data.

Locking paradigms–

1. Open the entire DB in exclusive mode.

2. Lock all records in the table being modified.

3. Lock the record currently being edited.

4. User specifies no locks – it is upto the software.

6. Open a table in Read Only mode.

Deadlock

User 1 has X lock on record 1	User 2 has X lock on record 2
User 1 requests Read on record 2	User 2 requests read on record 1
Waits	Waits

Avoidance : 1. Abort one

2. ensure that records are modified in same sequence. No user calls up record 2 before record 1.

Security

Basics – groups, users, userid, password, encyption.

Access to specific objects in the DB.

Open Systems (ODBC)

Open systems allow for portability

VB to Oracle, Sybase is possible using ODBC drivers. Excel can also use ODBC to connect to Oracle and store results within itself.

Client Server Computing

Expensive computer can be made available to a lot of users

Client stations can update the data

Consistency is maintained because only one copy of data is preserved

The DB processing is done by server, with client only displaying the results

Communication time between DB and Client is minimized because only results are sent back.

Object Oriented Databases

Object Oriented databases were designed so as to be able to store Objects. Not only numbers and characters but also drawings, images, photographs, video,voice.

Chapter 51 Database design and Normalisation

Diagram of DBMS in an office to get a reference point.

Brief of how a table looks like, how a structure of tables look like, how they are related. What is a relational database.

What is a key, primary key, foreign key.

Normalisation : What is ? it is a process to come up with the best possible design for a database.

Why needed ? we will understand after we do the rules.

Example – Company, Department, Employee

First Normal Form :

Department and Employee in one table – not in any normal form.

Discuss why this is not good.

Design this in First Normal Form

Company Code, Department Code, Name, Emp Code, Empl Name, Emp DOB, Manager Code, Manager Name

Split the Department and Employee table. Department will have Employee

Number. Employee Table will also have Employee Number.

A table is in First Normal Form if it contains no repeating attributes or groups of attributes.

Second Normal Form :

A Link table is introduced. Employee Number in Department table gets moved there. Design this as a Many-Many relationship.

One Emp can be in more than one department, one department can hold more than one Emp.

Employee table still has Manager code.

Link table – Dept Num, Emp Num

Emp Table – Emp Num

Dept Table – Dept Num, Dept Name, Mgr Num, Mgr Name

A table is in second Normal Form (2NF) if it is in first normal form and contains no partial dependencies.

Third Normal Form :

The Mgr Name in the Dept Table is such that the column is dependent on the Mgr Code which is not a key. So there is a non key dependency.

So move that to a separate table.

Create an Emp to Manager Link table. One emp can have one Manager, one manager can have many employees. One to Many.

Is that separate table needed if it is a one to many relationship. Put Mgr code in the Emp table and remove the link table, keeping the separate Manager table.

A table is in third normal form (3NF) if it is in second normal form and contains no 'non-key dependencies'.

Boyce Codd Normal Form (BCNF)

This is not strictly followed in the Data base modeling world, but I would like to stress that this is equally important in db design and I do not want any of my students to design a table that does not follow this rule.

The Rule states :

When a table is in BCNF, every attribute which is not part of the primary key is a fact about the key, the whole key and nothing but the key – so help me Codd.

For a relation with one candidate key, 3NF and BCNF are equivalent.

A table is in BCNF if and only if every determinant in the relation is a candidate key.

Now we go back to

WHAT IS A RELATIONAL DATABASE

A collection of tables where relationships are modeled by shared attributes.

Foreign Key : an attribute in one table that is a primary key in another table.

Chapter 52 Querying a Database

SQL – Structured Query Language

Used to get information from a set of tables in a database, grammar of queries in computing, it is also used to input data into tables, create tables, maintain them, and so on.

Asking questions from the DB is called DML.

1. Select 1 col from 1 table.
2. Select 2 cols from 1 table.
3. Introduce Distinct.
4. Introduce Where clause
5. Introduce conditions in a where clause
6. Specify a sort
7. Group By when you need a SUM
8. Extracting from more than one table.

Chapter 53. Data Definition and Data Manipulation

DDL – a language to define the structure and instances of a database.

Create table Employee (Emp ID int not null primary key,

Name varchar(10) not null,

HiredDate DATE,

Salary CURRENCY)

DML – asking questions from DB is DML.

also, it allows for manipulation of data in the database.

Insert into employee (emp_id, name) values ('1122','Anand')

Update employee set name = 'Raj' where emp_id = 1122;

Delete from employee where emp_id = 1122;

Chapter 54 Analysing a System (do with open txt book)

- **Parallel to defining the project.**

Systems investigation

1. Feasibility study. Hardware feasibility. Availability of software. Technical know how. Tools. User acceptance. Maintenance requirements.
2. Information requirements. Complete understanding of the current system and how it will change. (read Systems Investigation for complete understanding)
3. Fact finding methods :

   (a) Observation of the task

   (b) Examine existing documents

   (c) Surveys/Questionnaires

   (d) Interviews

   Reporting techniques

   (a) Where the data originates

(b) What processing is done, by whom

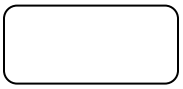
(c) Who used the data

(d) What data is stored and where

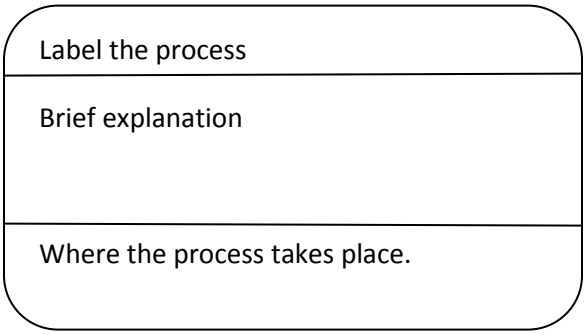
(e) What output is received and who uses it

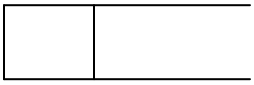
Data Flow Diagrams

External entity. Data source or data destination.

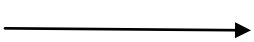


Process. An operation performed on the data.





Data Store. File, disk, tape



Data Flow

Levelled DFDs

Sometimes it is not possible to represent the complete business system in one diagram. Hence, 2 or 3 levels of data flow diagrams may be used, each showing more detail.

Class exercise (worksheet):

Students seek admission into Inventure school. Admissions office verifies from their records if the particular class has availability. If Available the student is accepted and course details are accepted into the DB. If not available a Rejection letter goes to the student.

Level one DFD : Only show In and Out items with Inventure System as a central DFD unit.

Level two DFD : Show complete Details in the diagram.

Entity Attribute Modelling :

The system (database) designer, DB Architect, produces a conceptual design identifying the various entities and attributes and showing how these are related. E-R diagrams are covered in Chap 22.

Data Dictionary :

It stores data about data. It stores names of data items (fields or variables), data types, length, validation criteria and other chars such as usage, physical representation, ownership, authorization and security.

Object-oriented Analysis Diagrams

When designing a system that is going to be programmed in an object oriented programming language, the analysis must include which methods act on these classes. See Chapter 38 for Class diagrams.

Volumetrics

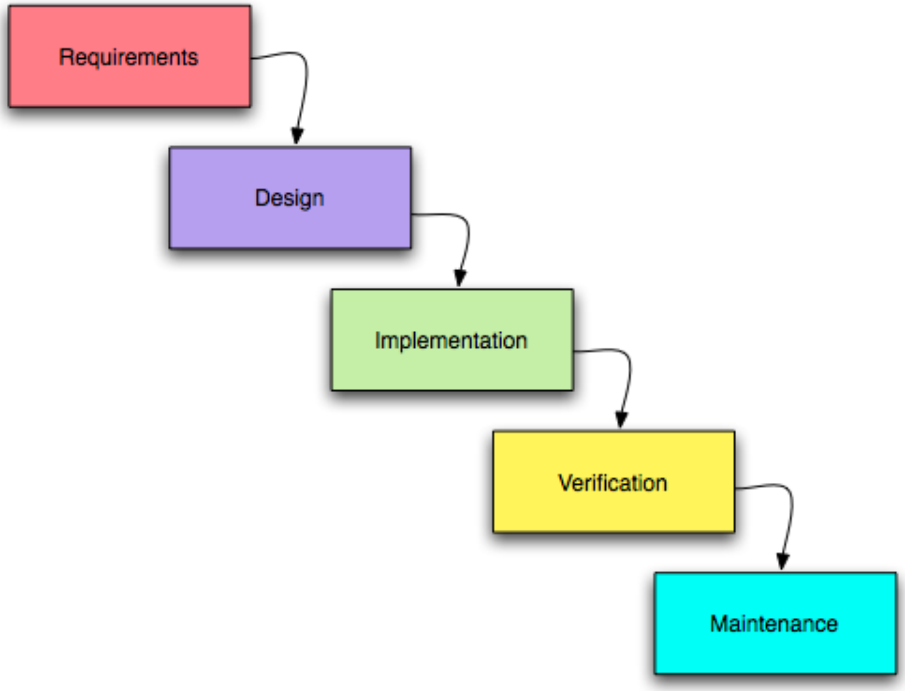
Volume of data affecting the system.

1. The number of input documents or on-line requests into the system.

2. Number of users and whether online or batch processing is required.

Chapter 55 – Systems Design, Development and Testing

The Waterfall Model to Systems design



Before u start a system –

Systems Designer will consider the following.

Output – content, format, sequence, frequency, medium

Input – volume, frequency, documents used, input methods

User interface – screens and dialogues, menus, special purpose requirements.

Type of System – batch, online, real-time

Files – contents, record layout, organization and access methods.

Processing – the programs and procedures needed and their detailed design

Security – how to secure the data from accidental corruption or deliberate tampering or hacking

Testing strategies – well tested before going live

Hardware – selection of an appropriate configuration.

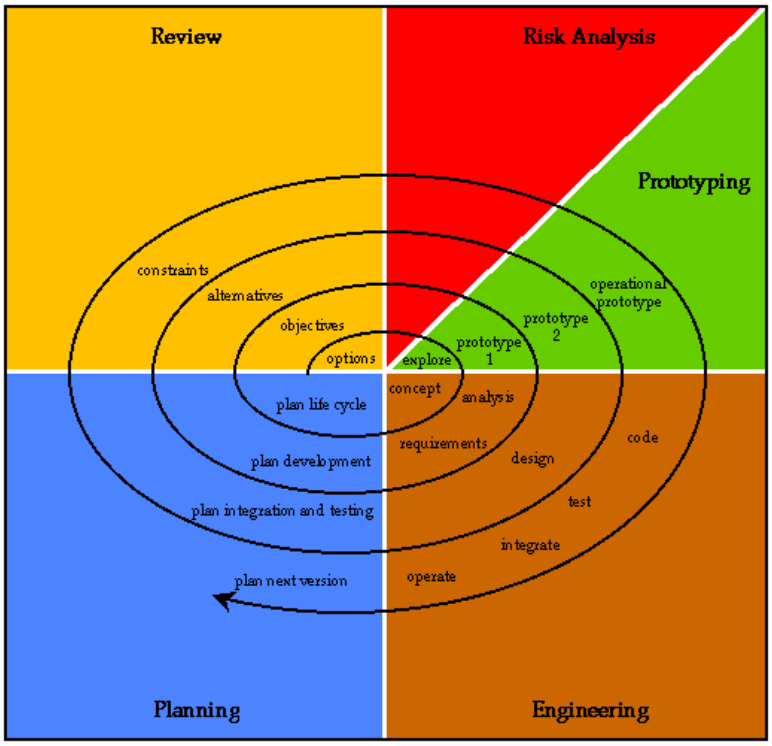
Prototyping

Prototyping is a useful design tool. It involves building a working model of a system in order to evaluate it, test it or have it approved before building the final product.

Throwaway prototyping – where prototypes are discarded before the real system is started.

Evolutionary prototyping – where prototypes are developed into a working system.

The prototyping model falls under the spiral model (as opposed to the waterfall approach to systems design).



Systems flowcharts

When a systems analyst is developing a new computer system, his or her ideas will need to be written down. A flowchart diagram showing the various tasks will go a long way in conceptualizing the system.

i. Flowchart would have :

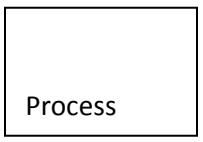
ii. Tasks carried out

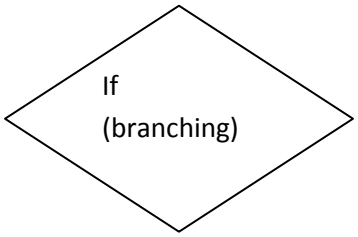
iii. Devices used (hdd, tapes,…)

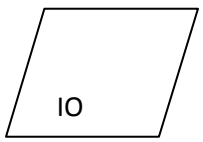
iv. Input output media

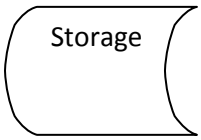
v. Files used

Symbols used :









User Interface

User Interface design refers the the technique of creating a screen that is easily used by the user, he can easily understand what is in it, what information it is requesting. It should direct the user on what to do next very clearly. It should clearly point out errors in data entry and let the user know where he has made a mistake.

Basically a good user interface design goes a long way in Systems acceptance and usability of the system.

1. Who is going to use the system
2. What tasks the computer is performing
3. What is technologically feasible.

Program design

This involves drawing hierarchy charts and structure charts and writing detailed program specifications. This is followed by algorithm design and pseudocode before coding.

Test Data

Select the data that you want to test with. Preparing for the test data at design stage is useful because you are thinking about the data as well when you are doing the design.

Development

Development includes coding and testing the program and testing the system as a whole.

Testing strategies

A system must be thoroughly tested before being installed. The designer should come up with a test strategy to make sure all parts of the system are properly tested.

Bottom up testing (we will use this)

Individual programs are tested using prepared test data, complete programs are then tested using data which ensures the whole system is covered, all routes are tested.

Top-down testing

The skeleton of the complete system is tested with individual modules becoming 'stubs' which just displays – executed.Testing each part of the system is sometimes referred to as unit testing.

Black box testing (functional testing)

Black box testing is carried out independently of the code used in the program. The system is considered to be a black box and test data is created from the specifications.

White box testing (structural testing)

This is dependent on the code logic and derives from the program structure rather than its function.

In white box testing you cannot test what isn't there.

Before release :

Unit testing

Module testing

Subsystems (collection of modules)

System/ Integration testing

Acceptance testing/User Acceptance Testing/User testing

Alpha testing : done by the software developers inhouse testing team.

Beta testing : giving the software (almost in completed stage) to a set of outside users who will now use the system as well as test and report any defects.

Chapter 56 Implementation, Evaluation and Maintenance

Implementation : This is a stage where people begin to use the new system. There are several tasks before the changeover is complete.

Install the hardware

Install the software

Creation of master files

Data for all master files will need to be entered before the new system can be used. Eg : Faculty data, Student data.

Choices when converting from an old system to a new one

Methods of conversion.

Direct changeover – *Adv*. No data duplication, fast and efficient

Disadv – if problems occur business could be disrupted.

Parallel conversion – *Adv.* Results from new system can be checked against known results. New system can be tweaked if errors are present without disruption to business.

Disadv. Duplication of effort could be an issue.

Phased conversion – Used with larger systems where different modules can be implemented in phases. Again Phased conversion could be direct or parallel.

Pilot conversion – where a small portion of the organization uses the new system.

Training for users.

Moving from old system to a new one would require additional user training. Detailed user manuals, documentation is finalized before system goes live.

Training staff is a crucial part of the success of any new system. Managers will need to be trained in using the system. They should feel confident in showing their staff to perform critical tasks.

Technical staff will need to be trained in correct backup procedures, customization for specific user needs, troubleshooting when things go wrong.

Manuals :

Installation Manual : how and what are the hardware requirements for installing the system. (P 330)

Operations Manual : How to operate the system for day to day things. Any backups required, reports to be generated, recovery processes.

User Manual : Using the system.Part of training requirements.

Post Implementation review :

1. Systems actual performance vs anticipated performance objectives
2. Assessment of each aspect of the system against preset criteria.
3. Errors which were made during System development.

4. Unexpected benefits and problems.

Software maintenance :

Three categories :

Perfective maintenance: The system could be made better in some way without changing its functionality. For example run faster or produce reports.

Adaptive maintenance: Changing needs in a company may mean changing system to adapt. For example a single user system can be changed to a multi user system.

Corrective maintenance : Involves correction of previously undetected errors. Commercial software programs like Word, Excel have bugs in them and maintenance releases are regularly brought out.

Factors affecting maintainability (can skip this)

1. Law of continuing change (programs in real world environment must change)
2. Law of increasing complexity (changes will make it more complex)
3. Law of large program evolution. Evolution is self regulating. System attributes such as time between releases, size and the number of reported errors are approximately invariant.
4. Law of organizational stability. Over the programs life, its rate of development is approximately constant.
5. Law of conservation of familiarity. Over the life of a system, the incremental change in each release is approximately constant.

Maintenance (Systems) documentation

1. Requirement specification

2. Hardware and software specification
3. Design diagrams such as DFDs, System flowcharts, E-R diagrams, Class diagrams
4. Overall system design showing the component parts and data structures.
5. Data dictionary
6. Algorithms
7. Clearly annotated program listings
8. Configuration settings.

Chapter 57 Input and Output Methods

Input devices

Scanner – used in scanning documents, photos. Can scan black and white as well as color. Resolution is measured in DPI, dots per inch.

Three passes is required for color scanning, one each for each filter Rewd, Green and Blue.

The scanner shines a bright light on the document and the scan head moves from top to bottom. The scan head passes over a line and collects data by measuring the intensity of light that is reflected back from the document.

Touch Screens – These are used in industrial environments such as manufacturing, wareohousing and security systems. Also avionics and medicing. And of course phones.

For every day work in the office it is less useful as it becomes very tiring to use the touch screen.

Different sensor technologies would be :

1. Responds to pressure.
2. A plastic screen on top of the existing glass surface. Conductive coatings are applied to the inner surface of both elements.

Magnetic Ink character recognition : **MICR**, is a character recognition technology used primarily by the banking industry to facilitate the processing of cheques. The technology allows computers to read information (such as account numbers) off printed documents. Unlike barcodes or similar technologies, however, MICR codes can be easily read by humans.

The use of magnetic printing allows the characters to be read reliably even if they have been overprinted or obscured by other marks, such as cancellation stamps and signature.

Optical character recognition, usually abbreviated to **OCR**, is the mechanical or electronic translation of scanned images of handwritten, typewritten or printed text into machine-encoded text. It is widely used to convert books and documents into electronic files, to computerize a record-keeping system in an office, or to publish the text on a website.

Optical Mark Recognition OMR : is used for grading answer sheets after exams. Students are asked to darken the circles with a pencil to mark the answer they choose.

Many traditional OMR (Optical Mark Recognition) devices work with a dedicated scanner device that shines a beam of light onto the form paper. The contrasting reflectivity at predetermined positions on a page is then utilized to detect the marked areas because they reflect less light than the blank areas of the paper.

OMR does not require a complicated pattern recognition engine unlike OCR.

Chapter 58 Networking Methods

When a network is formed using physical cables over shorter distances they form a LAN. When a network comprises a wide geographical area and uses telephone lines or specialized digital communication lines they are part of a WAN.

Modes of Network Operation :

Baseband : Carries one signal at a time. A bit value of a 0 or a 1 is sent by the presence or absence of a voltage in the cable.

For short distances only. Like LAN.

Bandwidth is dedicated to one data channel. Very fast.

Broadband : Carries multiple signals on a fixed carrier wave with the signals for 0 and 1 sent as variations on this wave.

WANs usually use broadband. It is faster than using a telephone line and modem. Simultaneous transmission of video, voice and data is possible in broadband.

Synchronous data transmission :

Timing signals are used to transmit data. Start and stop bits are NOT needed. Speed is higher because of this, but there may be more errors. Used for LANs usually.

Time Division Multiplexing :

A multiplexor joins more than one data stream into a single stream of data that can be transmitted over a communication channel. At the receiving end, the demultiplexor separates the single stream of data into its separate data streams.

Communication happens with time slots being transmitted. Time slots can be transmitted with or without data. Data is secondary, time slots are sent regardless.

In TDM many data streams share a single line. Each data stream is assigned to a time slot. Many users can thus carry on conversation simultaneously on a TDM line. A TDM line which caters to 3 data streams must be at least 3 times faster than the 3 lines combined – otherwise whats the point.

Circuit Switching :

When a caller dials a number, the path between the two telephones is set up by operating all of the exchanges involved in the path.The circuit remains on continuously for the whole call.

The two devices must transmit and receive data at the same rate. There is no time or resource wastage to reconstruct messages.

A leased line can be used if electrical interference produced by switching is a hindrance to data transmission between 2 computers.

Easier to hack into and listen in on calls, electrical interference can be a problem for data transfers using circuit switched networks, PSS is cost effective as cost depends on number of packets sent, more likely to be affected by network failure as PSS can send along different route.

Packet switching :

In a packet switching system messages are divided into packets of 128 bytes each. Each packet contains

1. Source and destination address
2. A packet sequence number so that the whole message can be correctly reassembled.
3. A checksum for the purpose of error checking.

Datagrams :

In TCP/IP a packet is called a datagram. The network takes a packet and redirects it down to the next node along its way to the destination.

The Packet Switching system (PSS) ensures that all the packets are re-assembled correctly at the destination.

The internet is a PSS system. PSS can also store packets until the node is ready to send it, it can also do some error checking and correction whereby when a node has an error it is possible to correct the error and retransmit.

Virtual Circuits :

A PSS that establishes the route between sender and receiver before transmission of data packets start.

Example : If computer A wants to send a message to computer B, it first sends a Call Request packet to the first node. This node decides which node to route the request to and the next node repeats the routing decision until the request arrives at Computer B. Computer B now sends a Call Accept packet that retraces the route of the call request packet. Computer A can now transfer the message to computer B through this established route.

Packet Switching vs Circuit Switching :

More efficient use of lines.

Cost depends on the number of packets sent, not on distance.

Less likely to be affected by network failure.

Better security. Data is less likely to be intercepted because packets can be sent along different routes or be interleaved with other unrelated packets.

Asynchronous Transfer mode :

ATM : a packet switching service that transmits data in short, fixed length packets referred to as cells, using a virtual circuit.

Standard protocols :

Protocol : pre agreed signals, codes and rules to be used for data and information exchange between systems.

A Standard Protocol is a protocol that conforms to a standard laid down by a standards authority to allow data exchange between any computer

Chapter 59 Local Area Networks

Local area networks are a set of links that connect together computers that are geographically close.

Type of cabling :

Twisted pair – like the ones found in telephone cabling

Co-axial – is high quality, well insulated cable which can transmit data much faster and more accurately than twisted pair.

Other communication methods :

Radio waves

Microwaves – stations should be not more than 30 miles apart because of the earth's curvature and microwaves travel in a straight line.

Communications satellites.

A computer is connected to a network by a network interface card which must match the type of network it is connected – like Ethernet, Token ring and ATM.

Network Topology :

How best to arrange the computers in a network has to be thought of before building a network. Different topologies offer different advantages and disadvantages.

Star Network. Eg : There is a mainframe computer in an office that other computers need access to. There is a main server which is very powerful and has an application software which is accessed by different programmers in the office.

Advantages : 1. Cable failure affects one computer only.

2.Consistent performance in times of heavy usage.

3. More secure. Messages cannot be intercepted by other stations.

4. Different stations can transmit at different speeds.

Disadvantages : Costly because of cable lengths.

Distributed star – will have many stars and connected to each other with connection boxes.

Bus network – All PCs share a common cable. Works well if the channels are not too heavily loaded.

Advantages : 1. Easy and inexpensive to instal.

2.Easy to add more stations without disrupting the network.

Disadvantages : Main cable goes down, whole n/w goes down.

Network performance degrades when there is heavy load.

Example : Ethernet

Ring Network – Computers are connected together. There is no central computer and no common cable. Messages are passed around the ring in one direction only.

Advantages – There is no dependence on a central computer or file server and each node controls transmission to and from itself.

Transmission of messages around the ring is relatively simple with messages travelling in one direction only.

Very high transmission rates are possible.

Disadvantages : If one node breaks down, n/w breaks down.

Example : Token Ring.

Operation of the networks

Bus (open book for diagram, draw on board)

All computers are connected together in a straight line. When a computer has data to send, it is first addressed, it then breaks it down into packets, and sends it across the network as electronic signals.

The computer to which it is addressed will accept the data.

Note that this system may have several stations sending data at the same time. This results in a collision which has to be resolved. One method is called CSMA-CD in an Ethernet. CSMA-CD stands for Carrier sense multiple access with collision detection. Here, the computer checks to make sure channel is not busy before sending data. If busy, it has to wait before sending data.

Once it begins transmission it has to listen to other nodes before beginning transmission. If there is a collision both stations will have to wait a random period before trying again.

What is a segment : It is a run of cable to which are attached a number of workstations.

What is a bridge : It connects two segments in a network. A bridge will memorise addresses and assign a frame to the appropriate sement. It will also amplify the signal so that the network is faster.

What is a hub : It is used to connect together computers in a network in a star fashion.

What is a switch : Intelligent hub used to directly deliver data to another computer on the network.

What is a Repeater : A repeater is used to amplify signals when the cables get lengthy. This takes some time, so care needs to be taken to make sure there are no network data drops.

Ring Network – Computers are connected in a ring. Signals travel in one direction only in a ring network.

A small packet called a token is passed to each computer. If the computer has information to send, it will modify the token, add address information and sends it down the ring. The information passes around the ring until the token reaches the destination.

Disadv : If one node fails, the whole network will go down.

STAR

Here computers are connected to each other through cable segments to a central hub.

When a signal is sent from a computer it is received by the hub and then sent across each segment to the other computers connected to the hub. Only the computer it is addressed to acts upon the data.

When a computer in a star network fails it does not affect the other computers except when the failure occurs in the hub.

Switched Ethernet : Ethernets have a maximum length for operation. 500 m. Hubs/repeaters can be used to extend the Ethernet and then the Ethernet can extend to a maximum of 2500 mts.

Switches are intelligent hubs which deliver packets directly to the destination without having to try delivering to each computer in the network. This also eliminates collisions.

Server based network. A network where resource security and administration and other functions are provided by dedicated servers.Windows 2000 Server.

Peer to peer networks. A network where all computers have equal status. There are no dedicated servers.MS Windows 2000.

From the Notes :

LANs use digital signals to transfer data between nodes. WANs can use digital signals in a local network but will use the analogue signals to transmit over the internet.

Different types of cables are used for different purposes. Coaxial, Shielded Twisted pair and Unshielded Twisted pair are some types of cables used in network connections.

SWITCHES are used for wired networks. Higher performance. Not for WANs.

ROUTER : Router receives the network level packet, network destination address is checked and the packet is then sent to the appropriate network. Routers can also enclose the packet in another packet and initiate the transfer through a new route by assigning another address.

BRIDGES : These connect two segments in a network environment. Bridge learns where the nodes are located and will send the packet to another node if necessary or the comp to which the packet is intended if the comp is on the same segment.

Chapter 60 Wide Area Networks

They connect together geographically remote computers.

2 options of communication over a WAN

Dial-up networking : User connects to the WAN using a standard telephone network.

Dedicated or leased lines : User's computer is permanently connected to the WAN.

Value added networks :

A privately owned wide area network that provides a specific service, not readily available on public networks but managed by a third party. (Like Cloud Computing).

The VAN is set up by a firm that is charge of managing the network. They then sell subscriptions to other companies, and charge a fee to use the network.

Electronic Data Interchange :

Transmission of business data from one computer system to another computer system via a wide area network.

Security

Keys

Encryption

Connecting to a wide area network

1. Use of a dial up telephone line and a modem. Modem speeds vary and is typically 56Kbps.
2. ISDN – modem is not required as the line is digital in itself. A modem is not required, instead a network termination device and a terminal adapter are required. These are commonly called digital modems. They use baseband transmission.
3. Another version is called B-ISDN, uses broadband transmission and is able to support transmission rates of 1.5.Mbps. B-ISDN required fibre optic cables which is not widely available.
4. Cable Modem : Cable companies now offer Internet services on broadband transmission using something called cable modems. A cable modem can transfer data at 500 kbps or higher, compared with 28.8 to 56 kbps for common telephone line modems.
5. ADSL – Asymmetric Digital Subscriber Line is a technology which turns telephone copper wires into 'fat data pipes'. This means that existing telephone lines can provide broadband transmission. This requires a special ADSL modem.

   The "asymmetric" in ADSL refers to the fact that the downstream data rate, or the data coming to your computer from the Internet, is traveling faster than upstream data, or the data traveling from your computer to the Internet. Upstream data rates are slower because Web page requests are fairly

miniscule data strings that do not require much bandwidth to handle efficiently.

6. CODEC – a device that encodes or decodes a signal and may compress and decompress these signals in the process.
   For example : CODEC can be used to compress video from a video camera so that it can be stored and played on a computer.

Somethings More

Baseband : A baseband signal or lowpass signal is a signal that can include frequencies that are very near zero. Used for Ethernets, LANs. Very fast.

Broadband : A signaling method which can include or handle a wide range of frequencies.

Compare broadband/baseband

Broadband is like cable television. Multiple channels can be submitted simultaneously in Broadband. Used when sending large amounts of data over a wide area where wait times can be unacceptable.

Baseband signaling would be sending only a single signal over the cable. This type is used in Ethernet networks. They are simple and fast.

Chapter 61. Internetworking

Internetwork is the connection between networks. Best known internetwork is an Internet.

Routers : a device that forwards packets between networks using IP addresses.

Routers use headers and forwarding tables to determine the best path for forwarding the packets and they use protocols such as TCP/IP to communicate with each other and configure the best route between any two hosts.

It uses a combination of network and node address to communicate with each other.

Gateways : a device used to connect networks using different protocols. It repackages the data so that it can be read correctly at the receiving end.

It is also called a protocol converter.

A gateway acts as a conversion from one protocol to another or in the case of VoIP from the VoIP network to the POTS network. A router works by looking at the IP address in the packet and decides if it is for internal use or if the packet should move outside the network. In a VoIP conversation, you would have a gateway that works for any calls that cannot be carried via IP and it moves them to a POTS connection. You generally need both.

TCP/IP

TCP is a set of rules used to perform handshake, packet sequencing, flow control and error handling.

IP is a set of rules that are used to send packets from one network to another using IP addresses.

IP address : is an address given to a machine on a network in a place.

Non routable addresses are for intranets.

Subnet masks : These are used to determine the network and the host parts of an address. It is a 32 bit number which when ANDed to the IP address will be used to determine which part is the Host and which one is the Network.

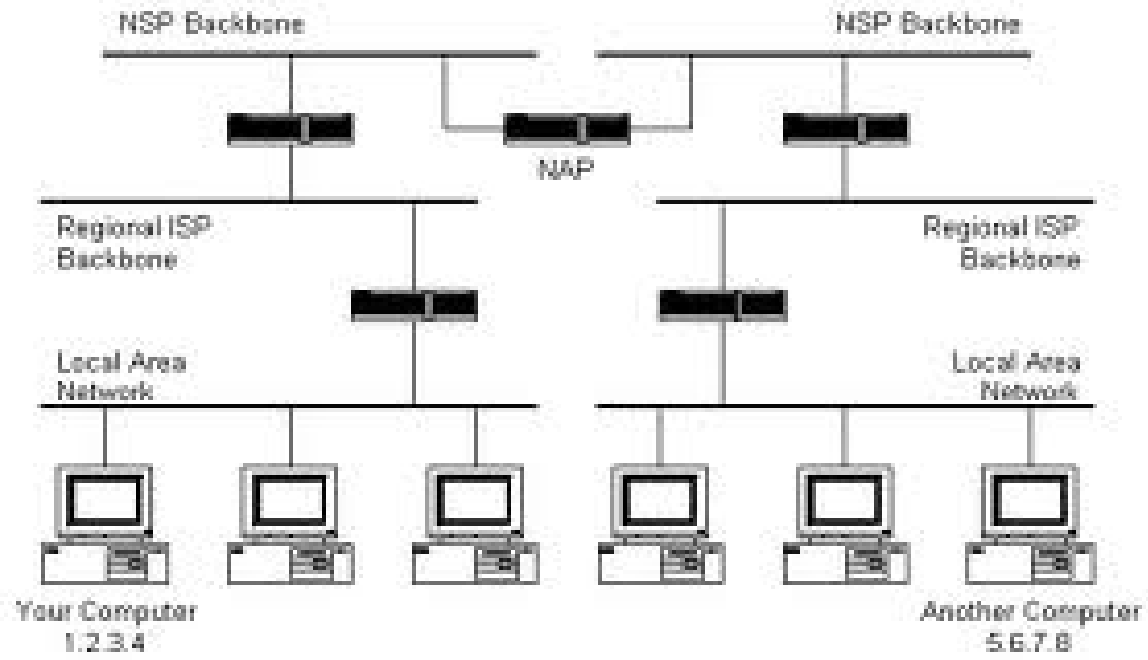
If two computers A and B's IP addresses are like this :

192.168.0.4 and 192.168.0.5 then the subnet mask would be

255.255.255.0 which means both the computers are on the same subnet and hence a router will not be needed to communicate between the two comps.

Chapter 62. The Internet

A computer is typically connected to an ISP which in turn connects to the backbone.



When a packet arrives at a router, the router examines the IP address put there by the IP protocol layer on the originating computer. The router checks it's routing table. If the network containing the IP address is found, the packet is sent to that network. If the network containing the IP address is not found, then the router sends the packet on a default route, usually up the backbone hierarchy to the next router. Hopefully the next router will know where to send the packet. If it does not, again the packet is routed upwards until it reaches a NSP backbone. The routers connected to the NSP backbones hold the largest routing tables and here the packet will be routed to the correct backbone, where it will begin its journey 'downward' through smaller and smaller networks until it finds it's destination.

Chapter 64. HTML and ASP

What is FTP

FTP works by opening up a channel for communication and then staying open till you end the session. You then type in commands to get files from the site and/or put files into the site. Someone opens up their server for the world to communicate **textually** with them.

Open ftp.co.in (will open a ftp connection)

Typically you will enter the userid and password after doing this.

If it is an anonymous ftp site – no user id and password will be asked.

Then u will type in commands like put and get or mput and mget to work with files.

Exit will end the session.

Go to this anonymous ftp site to browse : ftp://www.imtech.res.in/

What is HTML – Hypertext Markup Language

Basically it is a language of tags. You need to memorize tags in order to build and HTML document.

To include logic into an HTML document you will use Javascript or Active Server Pages (from Microsoft)

What do u need to build a web page and show it to the world. You need to upload it into a server provided by a third party or build your own server and host it there.

```
<html>

03  <head>
04
05  <title>Winter</title>
06
```

```
</head>
<body>

<h1>Winter Vacation</h1>


<p>My winter vacation was cold, chilly, and a lot of fun on the snow.</p>
<p>How many days till Spring?</p>
</body>

</html>
```

OR

```
<html>
<head>
<title>My Web-page</title>
</head><body bgcolor="yellow">
I love wikiHow because
<marquee>It's the best website in the world!</marquee>
<img
src="http://www.wikihow.com/skins/common/images/wikiHow_logo_5.gif"></i
mg>
</body>
</html>
```

Web Server diagram. ASP scripts are executed in the server. Server side scripting languages include Javascript, Perl, VB Script and ASP.net.

You build a form, put controls into the form, use scripting to deal with data and save it into the database and retrieve and show data on forms from database. Then you save the form on a Web Server in a virtual directory. The web server will handle all details on loading the form and handling the form for use in an application.

ASP script example :

```
<html>
<body>
<form action="demo_simpleform.asp" method="post">
```

```
Your name: <input type="text" name="fname" size="20" />
<input type="submit" value="Submit" />
</form>
<%
dim fname
fname=Request.Form("fname")
If fname<>"" Then
    Response.Write("Hello " & fname & "!<br />")
    Response.Write("How are you today?")
End If
%>
</body>
</html>
```

POST and GET

Post will send data to server and will be invisible to the user as request is sent via an header.

Get will send data and will be visible to the user as request will be sent as part of URL.

Use GET:
- during development for debugging purposes (although in ASP.NET it's also easy to see what has been sent through POST).
- if you want your visitors to be able to bookmark the submitted pages
- if you want to refer to submitted pages using hyperlinks

Use POST:
- for forms with password fields
- for large forms or forms with large text fields

Please note that web forms in ASP.NET use POST by default.

It can be changed into GET, but only for small forms.

Telnet

Telnet is used when you want to get into another computer and work on it remotely.

It is a terminal emulation program.It is used to remotely control servers. When you work from home you will use this a lot. You can also control servers in the US from a remote location like India or vice versa.

Chapter 65 Java and Applets

Java is a language derived from C++ and loses the pointer – the dreaded data reference in C++ ☺

Applets are written particularly for the internet. An applet call can be embedded into HTML code, the applet will execute on the client.

Browsers contain a JVM which help in executing applet byte code. Applets do not/cannot change anything on the computer so they are very safe.

<APPLET CODE = “Hello.class” WIDTH=200 HEIGHT=100> </APPLET>

A window will open with the size specified and the applet will execute within that window.