Problem Solving Ability

Module 1:-

1) What is the difference between script and program?

A "program" in general is a sequence of instructions written so that a computer can perform certain task. [Or, A program is usually a standalone compiled executable in its own right (although it might have library dependencies), consisting of machine code or byte codes (for just-in-time compiled programs).

A "script" is code written in a scripting language. A scripting language is nothing but a type of programming language in which we can write code to control another software application. Scripts are usually interpreted.

Date __/__/____

2) What are the basic components of a digital software?

A typical digital computer has four basic functional elements:

i) Input - output equipment
ii) Main Memory
iii) Control Unit
iv) Arithmetic & Logic Unit.

3) Differentiate between compiler and interpreter.

Compiler	Interpreter
i) Compiler scans the program in one go.	i) Translates program one statement at a time.
ii) As it scans the code in one go, the errors (if any) are shown at the end together.	ii) Considering it scans code one line at a time, errors are shown line by line.
iii) Main advantage of compilers is its execution time.	iii) Due to interpreters being slow in executing the object code, it is preferred less.
iv) It converts source code to object code.	iv) It does not convert source code into object code instead it scans it line by line.
v) It does not require source code for later execution.	v) It requires source code for later execution.
Ex:- C, C++, C# etc.	Ex:- Python, Ruby, Perl, SNOBOL, MATLAB etc

Page No.

Date __/__/____

Saathi

4) Is RAM absolutely volatile?

Most RAM is volatile. If you turn off the power, the contents are lost.

But there is non-volatile RAM, but its always explicitly referred as NVRAM, for example, FRAM (Ferroelectric RAM), which has a layer of ferromagnetic material in its cell structure that maintains its alignment when power is off.

5) What is the basic role of operating system?

An operating system is the most important software that runs on a computer. It manages the computer's memory and processes, as well as all of its software and hardware. It also allows you to communicate with the computer without knowing how to speak the computer's language. Without an operating system, a computer is useless.

6) Differentiate between Secondary Memory (e.g. HDD), Primary memory (e.g., RAM), Cache Memory, and register?

Primary Memory	Secondary Memory	Cache Memory	Register
i) Primary Memory is temporary.	i) Secondary Memory is permanent.	i) Cache is a smaller and fastest memory component in the computer.	i) Register is a small amount fast storage element into the processor.
ii) Primary memory is directly accessible by processor/ CPU.	ii) Secondary memory is not directly accessible by the CPU.	iii) CPU accesses memory at the faster rate than register.	ii) CPU can operate on register contents at the rate of more than one operation in one clock cycle.

Page No.

Saathi

Date __/__/____

iii) Primary memory are more expensive than secondary storage devices.	iii) Secondary memory devices are less expensive when compared to primary memory devices.	iii) Cache memory is exactly a memory unit.	iii) It is used to store and retrieve located in the CPU.
iv) Nature of parts of Primary memory varies, RAM is volatile in nature, ROM is non-volatile.	iv) It is always non-volatile in nature.	iv) It is a high-speed storage area for temporary storage.	iv) It is used to store data temporarily for processing and transfer.

Module 2:-

1) Convert the Hexadecimal number 9AC2 to the corriesponding decimal number.

$(9AC2)_{16} \rightarrow (?)_2 \rightarrow (?)_{10}$

$(9AC2)_{16} \rightarrow (?)_{10}$

$9 \times 16^3 + 10(A) \times 16^2 + 12(C) \times 16^1 + 2 \times 16^0$

$= 36864 + 2560 + 192 + 2$

$= (39618)_{10}$

Date ___/___/_____

Saathi

2) Explain r's complement and (r-1)'s complement with example.

The r's complement of a non-zero number in any number system with base r can be found out by subtracting every single digit of a number by r.

The (r-1)'s complement of a number in any number system with base r can be found out by subtracting every single digit of a number by r-1.

3) Subtract $(1011)_2$ from $(1101)_2$ using 1's complement method.

$(1011)_2 - (1101)_2$

$\Rightarrow (1011)_2 + (0010)_2$

$$\begin{array}{r} 1011 \\ +0010 \\ \hline 1101 \end{array}$$

∴ Answer = $(0010)_2$

Subtract $(1011)_2$ from $(1101)_2$ using 1's complement method.

Module 3:-

1) State and prove De-Morgan's Theorem.

The first theorem of De Morgan's Law states that when two (or more) input variables are AND'ed and negated, they are equivalent to the OR of the complements of the individual variables.

$$\overline{AB} = \bar{A} + \bar{B}$$

The second theorem of De Morgan's Law states that when two (or more) input variables are OR'ed and negated, they are equivalent to the AND of the complements of the individual variable.

$$\overline{A+B} = \bar{A}\bar{B}$$

Proof:-

A	B	AB	$\bar{A}$	$\bar{B}$	$\overline{AB}$	$\bar{A}+\bar{B}$
0	0	0	1	1	1	1
0	1	0	1	0	1	1
1	0	0	0	1	1	1
1	1	1	0	0	0	0

∴ The truth table of $\overline{AB}$ & $\bar{A}+\bar{B}$ gives the same output, the Law stands.

Page No.

Date ___/___/___

A	B	$\bar{A}$	$\bar{B}$	$A+B$	$\overline{A+B}$	$\bar{A}\bar{B}$
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

Since, the outputs of $\overline{A+B}$ & $\bar{A}\bar{B}$ are same, hence the law stands.

2) What do you mean by Universal gate?

A universal gate is a gate which can implement any Boolean function without need to use any other gate type. The NAND and NOR gates are universal gates.

3) Explain why NAND and NOR gates are known as universal gates.

The NAND and NOR logic gates are called the universal logic gates since all the other remaining logic gates can be derived from them. They are very important as they can perform all the binary operation as well as also implement any of the Boolean functions.

4) What is a truth table?

A truth table is a tabular representation of all the combination of values for inputs and their corresponding outputs. It is a mathematical table that shows all possible outcomes that would occur from all possible scenarios that are considered factual, hence the name. Truth Tables are used for logic problems as in Boolean algebra and electronic circuits.

Page No.

Date __/__/____

Saathi

7) For the N variable Boolean function, what is the total number of rows/tuples in a truth table?

For N- variable boolean function, total number of rows in the truth table is 2^N.

The total number of literals is also N.

8) For 1024 literals, how many variables are there?

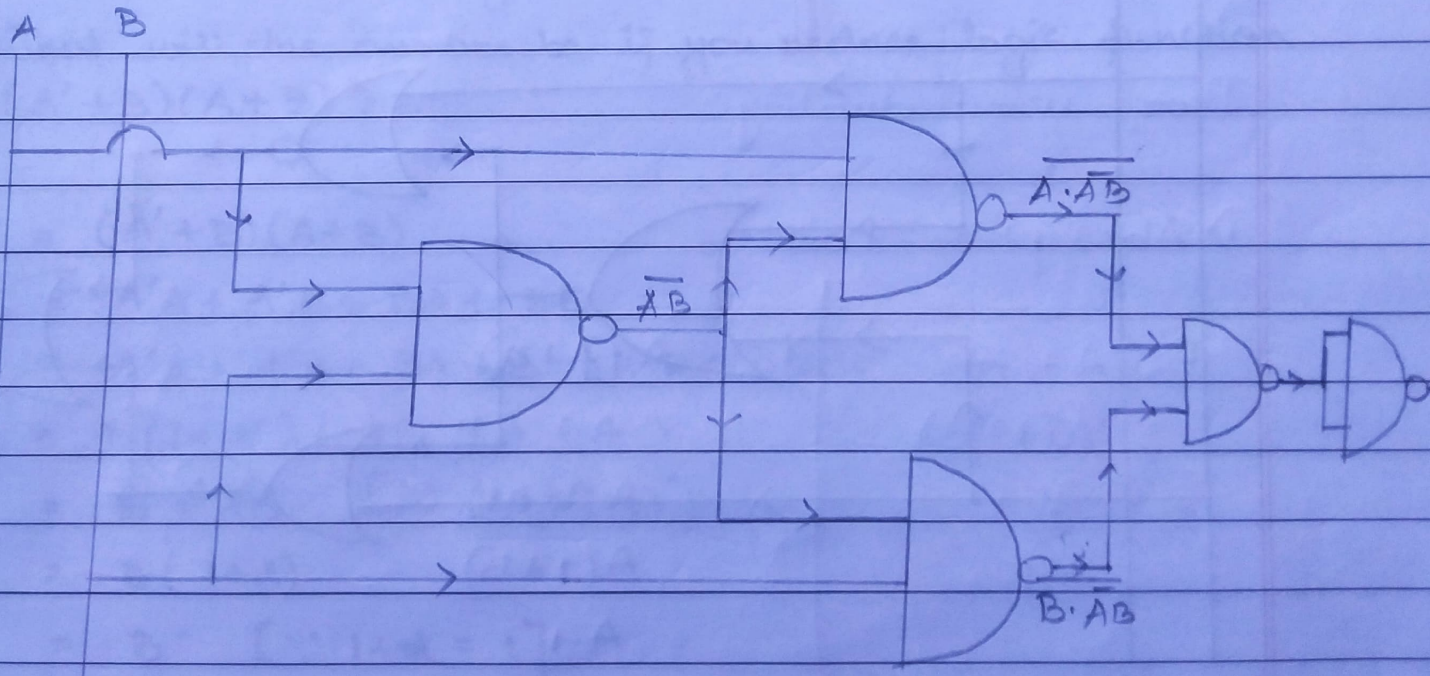
For 1024 literals, 10 variables are there.

$(2^{10} = 1024)$

9) Design XNOR gate using only NAND gate.

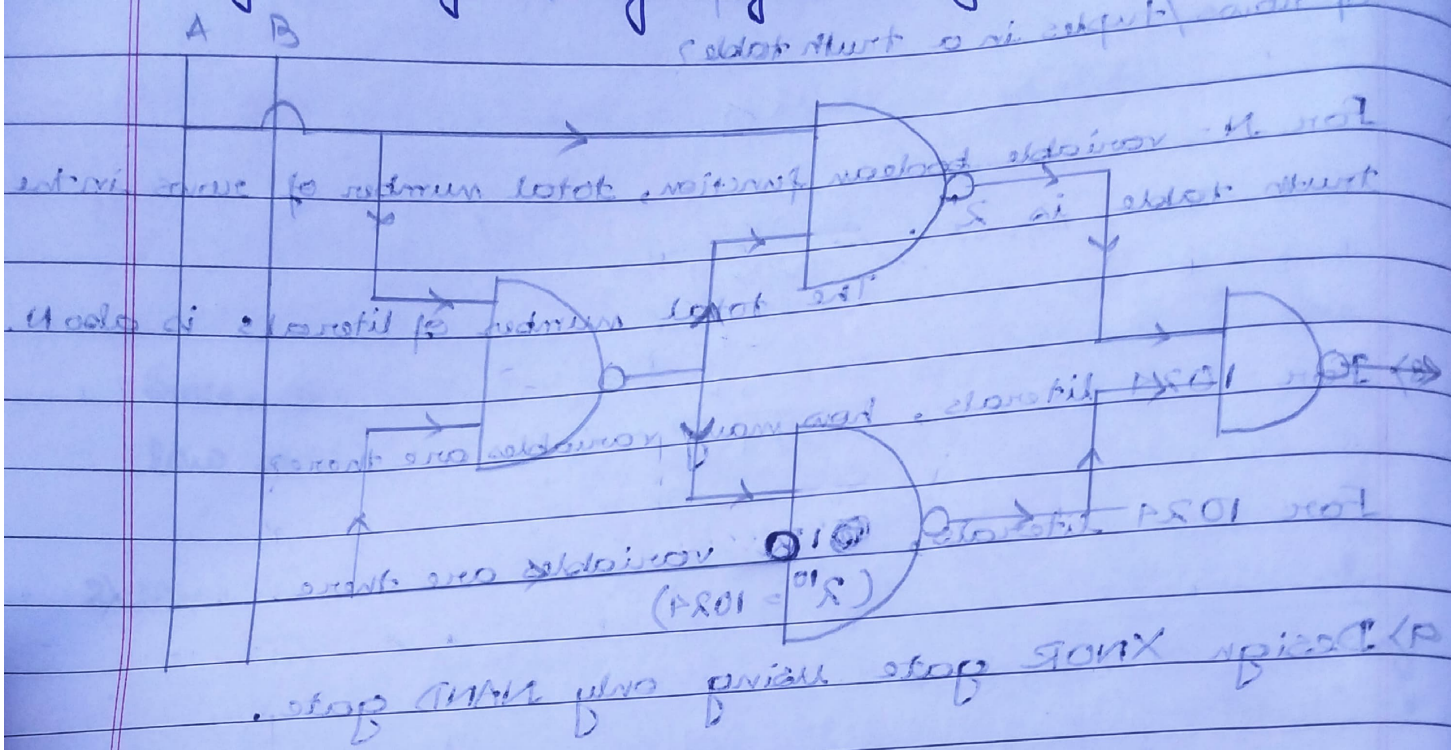
We know,

$Y = A \text{ (xnor) } B = (A'B' + AB)$

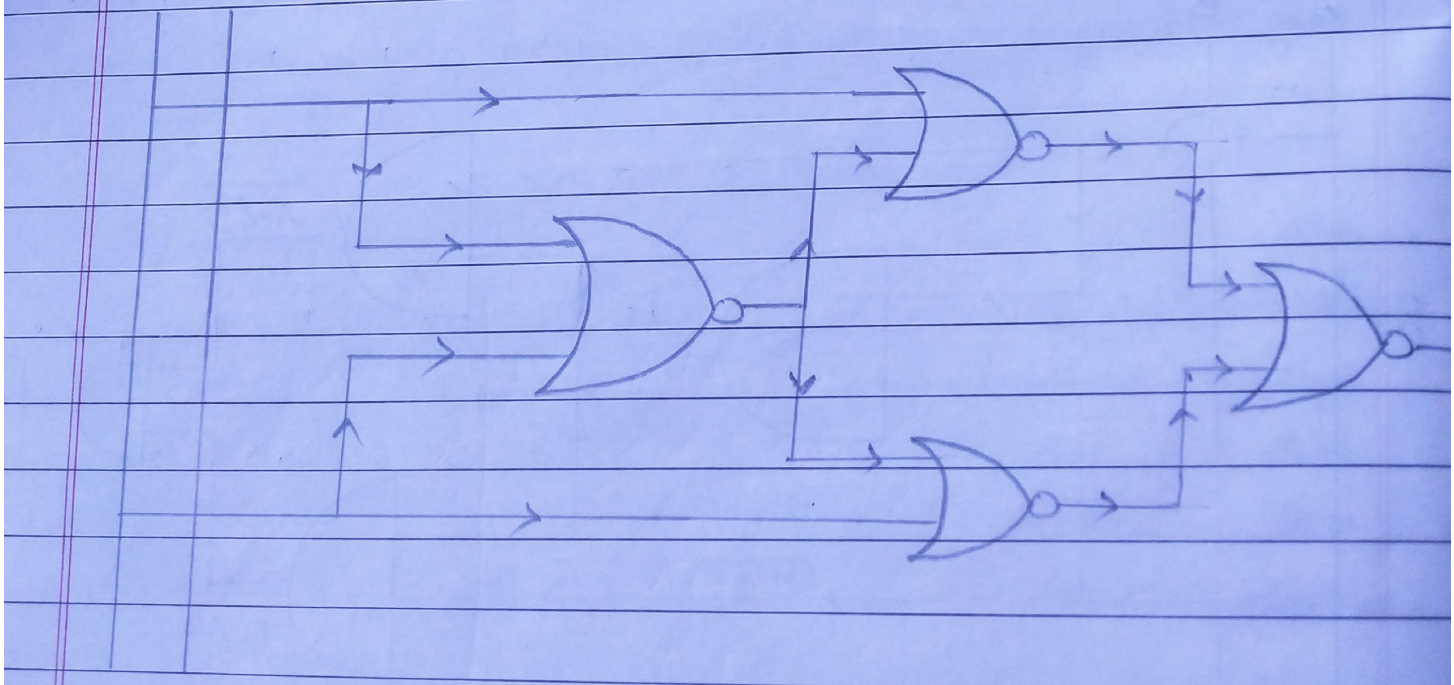


Date ___ / ___ / _____

10) Design XOR gate using only NAND gate?



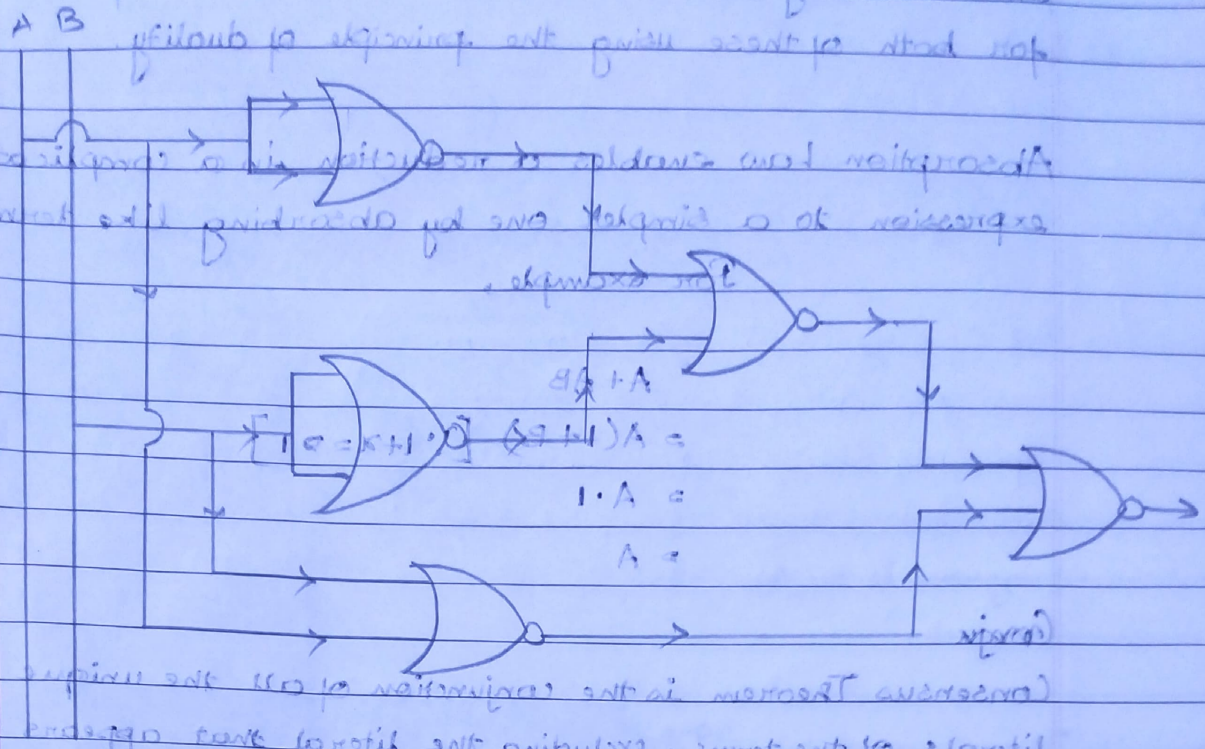
11) Design XNOR gate using only NOR gate



Date __/__/____

Saathi

12) Design XOR gate using only NOR gate.



13) What will the answer be if you reduce logic function $(A'+B)(A+B)$?

$$F = (A'+B)(A+B)$$
$$= A'A + A'B + BA + BB$$
$$= A'A + A'B + BA + B \quad [\because x \cdot x = x]$$
$$= B(1+A') + AB$$
$$= B + AB \quad [\because (1+x) = 1]$$
$$= B(1+A)$$
$$= B \quad [\because 1+x = 1]$$

14) What is the principle of duality related to Boolean Algebra?

The principle of duality in Boolean Algebra states that if you have a true Boolean statement (equation) then the dual of this statement (equation) is true. The dual of a boolean statement is found by replacing the statement's symbol with their counterparts.

Date __/__/____

Saathi

15) Explain absorption law and consensus law theorem in Boolean Algebra. Obtain the alternative expression for both of these using the principle of duality.

Absorption Law enables a reduction in a complicated expression to a simpler one by absorbing like terms.

For example,

$$\begin{aligned} & A + AB \\ &= A(1+B) \quad [\because 1+x = 1] \\ &= A \cdot 1 \\ &= A \end{aligned}$$

Consensus

Consensus Theorem is the conjunction of all the unique literals of the terms, excluding the literal that appears unnegated in one term and negated in the other.

Proof using duality:-

i) Absorption Law:-

$$\begin{aligned} F &= A + AB \\ &= A(1+B) \\ &= A \cdot 1 \\ &= A \end{aligned}$$

$$\begin{aligned} F' &= A \cdot (A+B) \\ &= A \cdot A + AB \\ &= A + AB \quad [\because] \\ &= A(1+B) \\ &= A \cdot 1 \quad [1 = A+1 \ \because] \\ &= A \end{aligned}$$

ii) Consensus Theorem:-

Consensus Theorem is $AB + \bar{A}C + BC = AB + \bar{A}C$.

Dual of Consensus Theorem is $(A+B)(\bar{A}+C)(B+C) = (A+B)(\bar{A}+C)$

Date __/__/____

Saathi

Therefore,

$[A+B][\bar{A}+C][B+C]$

$= [A\bar{A} + AC + B\bar{A} + BC][B+C]$

$= [AC + \bar{A}B + BC][B+C]$

$= AC \cdot B + AC \cdot C + \bar{A}B \cdot B + BC \cdot B + BC \cdot C$

$= ABC + AC + \bar{A}B + \bar{A}BC + BC + BC$

$= ABC + \bar{A}BC + AC + \bar{A}B + BC$

$= BC[A + \bar{A}] + AC + \bar{A}B + BC$

$= BC \cdot 1 + AC + \bar{A}B + BC$

$= BC + BC + AC + \bar{A}B$

$= BC + AC + \bar{A}B$

$= AC + \bar{A}B + BC$

$\therefore (A+B)(\bar{A}+C)(B+C) = AC + \bar{A}B + BC$

$(A+B)(\bar{A}+C) = A\bar{A} + \bar{A}B + AC + BC$

$= \bar{A}B + AC + BC$

∴ We can say,

$(A+B)(\bar{A}+C)(B+C) = (A+B)(\bar{A}+C)$

16) For 2 variable XOR logic what is an arithmetic expression for output?

17) For 2 variable XNOR logic what is an arithmetic expression for output?

Exor $xy' + x'y$

Exnor $xy + x'y'$

Page No.

Date __/__/____

Module 5:-

1) Differentiate between algorithm, program and flowchart. Give suitable example to explain your answer.

Criteria	Algorithm	Flowchart	Program
Description	An algorithm is a procedure or set of rules that defines how a program is to be executed.	A flowchart is a graphical representation of the steps a program takes to process data.	A program is a set of instruction given to the computer by the user to do something useful.
Complexity	It is comparatively difficult to create and also a bit challenging to be understood by a layman.	It is easy to design and also very user friendly.	It is quite complex for any layman to understand because the entire instruction to the computer is written in the syntax of the programming language.
Method of writing	An algorithm does not include any sort of geometrical pattern.	It utilizes different types of geometrical shapes, symbols and pattern.	It requires the knowledge of the syntax or method of a specific programming language.
Scope of Usage	An algorithm is used in domains of mathematic & computer science.	A flowchart is used in documenting different disciplines to describe a process.	A code or program is the ultimate instruction given directly to the computer to do something useful.

Date ___/___/_____

2) Define algorithm and its properties.

An algorithm is defined as the sequence of instructions written in simple english that are required to get desired results. It helps to develop fundamental logic of a problem that leads to a solution.

The properties of Algorithm are:-

i) Each step of an algorithm must be precisely defined.
ii) An algorithm must contain blocks that will help to solve problems more efficiently and logically.
iii) It should accept a set of inputs and produce a defined output.
iv) It must be terminated after a finite number of steps.
v) It should be independent from a computer programming language.
vi) It should develop a platform for writing programs.

3) Define Time and Space complexity with an example.

Time complexity is a function describing the amount of time an algorithm takes in terms of the amount of input to the algorithm.

Space Complexity is a function describing the amount of memory (space) an algorithm takes in terms of the amount of input to the algorithm.

For example: C=0 for i in range (0,5): C=C+1 Here, the counter C variable calculates a time complexity.	For example, the number of variables used in a program calculates space complexity. A program using 3 variables will use more memory than a program using 2 variables.

Date ___/___/_____

Saathi

4) Explain best, worst and average-case time complexity with a suitable example.

Best Case Time Complexity can be defined as the minimum time taken by an algorithm to solve a problem. For example,

```
a = int(input("Enter a number="))
b = int(input("Enter a number="))
print("The greater number is=", max(a,b))
```

Average Case Time Complexity can be defined as the average time (between worst and best case) taken by an algorithm to solve a problem. For example,

```
a = int(input("Enter a number="))
b = int(input("Enter a number="))
if (a>b):
    print("The greatest number is=", a)
else:
    print("The greatest number is=", b)
```

Worst Case Time Complexity can be defined as the maximum time taken by an algorithm to solve a problem. For example:-

```
a = int(input("Enter a number="))
max = a
for i in range(0,1):
    b = int(input("Enter a number"))
    if (b>max):
        max = b
print("The greatest number is=", max)
```

Date __/__/____

Saathi

5) Define asymptomatic time complexity. Explain Big-O, Big-Omega, Big-Omega, and Big-Theta time complexities with examples.

The limiting behaviour of the execution time of an algorithm when the size of the problem goes to infinity is called Asymptomatic Time Complexity. This is usually denoted in Big-O notation.

BigO Notation is a way to measure an algorithm's efficiency. It measures the time it takes to run your function as the input grows. Big O is sometimes referred to as the algorithm's upper bound, meaning that it deals with the worst-case scenario.

Big-Omega notation is a formal way to express the lower bound of an algorithm's running time. It measures the best case time complexity or the best amount of time an algorithm can possibly take to complete.

Big-Theta notation is the formal way to express both the lower bound and the upper bound of an algorithm's running time.

6) Write a small iterative algorithm for finding the length of a given number (length denotes number of digits). Also, explain the time complexities for your algorithm.

```
n = int(input("Enter a number"))
while (n>0):
    c = c+1
    n = n//10
```

The time complexity of the algorithm is O(n), because

Page No.

Date ___/___/_____

the loop will continue to run as long as n does not become zero. (The length of digit can be 1, 2, ... upto n). Hence, the order of time complexity.

7) Find out the time complexity of the following algorithm (with explanation).

```
int i, j, k = 0
for (i = n/2, i <= n, i++)
{
for (j = 2, j <= n, j = j * 2)
{
k = k + n/2
}
}
```

~~Comparing the loop variables i and j, with each other.~~

i ~ j

The outer loop runs for n/2 times and the inner loop runs for $\log_2^n$ times. Therefore, the time complexity of the algorithm is $O(n \log \frac{n}{2})$.

Module 6:-

1) ~~How to print the following statement in python:~~

Date __/__/____

Saathi

Module 7 :-

1) Difference between 5/2 with 5//2.

5/2	5//2
i) The value returned after this operation is the actual quotient (both whole number and fractional part) as answer result.	i) The value returned after this operation is the absolute value of the quotient (only whole number part) as answer result.
ii) The datatype of the result is a floating point.	ii) The datatype of the answer result is an integer.

2) Differentiate between relational operators with logical operator.

Relational Operator	Logical Operator
i) A relational operator is a programming language construct or operator that tests or defines some kind of relation between two entities.	i) A logical operator is a symbol or word used to connect two or more expressions such that the value of the compound expression depends only on that of the original expressions and on the meaning of the operator.
ii) The datatype of relational operator is boolean (either True or False).	ii) There is no datatype of logical operator but it makes sure that two or more relations are logically connected.
For eg., > (greater than), < (less than), == (equality) etc.	For eg., AND, OR, NOT etc.

Page No.

Date ___/___/_____

Module 8 :-

1) Differentiate between break, continue and pass with examples.

break	continue	pass
The break statement is used to bring the control out of the loop when some external condition is trigerred.	The continue statement in Python returns the control to the beginning of the iteration. The continue statement rejects all the remaining statements in the current iteration of the loop and moves the control back to the top.	Pass is a null statement. The interpreter does not ignore pass statement, but nothing happens and the statement results into no operation. The pass statement is useful when you don't the implementation of a function but you want to implement in future.

3) What is a for-else statement in Python? Give suitable example.

Python allows the else keyword to be used with the for and while loops too. The else block appears after the body of the loop. The statement in the else block will be executed after all iterations are completed. The program exits the loop only after the else block is executed.

Date __/__/____

Saathi

For example;

```
for x in range(5):
    print("Iteration no {} in for loop".format(x+1))
else:
    print("Else block in loop")
print("Out of the loop")
```

Output:-

Iteration no 1 in for loop
Iteration no 2 in for loop
Iteration no 3 in for loop
Iteration no 4 in for loop
Iteration no 5 in for loop
Else block in loop
Out of loop

2) Differentiate between aliasing data, shallow copy, and deep copy with examples:

Aliasing Data	Shallow Copy	Deep Copy
In python programming, the name given second name given to a piece of data is known as an alias. Aliasing happens when the value of one variable is assigned to another variable because variables are just names that store references to actual values.	A shallow copy constructs a new compound object and then (to the extent possible) inserts references into it to the objects found original.	Deep copy is a process in which the copying process occurs recursively. In case of deep copy, a copy of object is copied in other object. It means that any changes made to a copy of object does not reflect in the original object.

Page No.

Date __/__/____

Saathi

A) Differentiate between for and while loop with examples.

For Loop	While Loop
i) The structure for 'for loop' is :- for (initialvalue, finalvalue, increment) : Statement	i) The structure for 'while loop' is "while (condition) : Statement.
ii) Iterates for a preset number of times.	ii) Iterates till the condition is met.
iii) In absence of a terminating condition in for loop, the for loop will run for infite number of times.	iii) In absence of a terminating condition, while loop shows an error.
iv) Initialization of the loop is done only once when the program starts.	iv) Initialization is done every time when the loop is iterated.
v) Used to obtain a result when the number of iteration is known.	v) Used to control the condition when the number of iteration is unknown.
vi) It is exit controlled loop.	vi) It is entry control loop.

Saathi

Date ___/___/_____

Module 4:-

1) How to convert an integer array into float array? (Give a coding example on implicit & explicit type conversion).

An integer array can be converted to floating point array using the command 'astarray'. The syntax of the command is :-

```
x = np.astarray(array)
```

where array is the integer array,
np is numpy module imported as np.

~~The conversion of implicit conversion is,~~

The example of implicit conversion is

```
x = 0
x = 5/2
```

Initially, x was storing an integer value, but later it started storing a double float value. This con type of conversion does not require any special command. Hence, the name.

The example of explicit conversion is:

```
x = 1.0
x = int(5//2)
```

Initially, x was storing an integer (float) value, but later it stored an integer. This is not possible because float has higher heirarchy than integer. So, it has to be done explicitly. This is known as explicit type conversion.