

Theoretical concepts of Operating System

1.What are the notations for representing algorithms?

Or Explain the Notations of algorithm. (QN. 2,3,4)

- Programming language, pseudo code, and flowchart are notations for expressing algorithms.

2.Define Programming language

- A programming language is a notation for expressing algorithms.
- Set of Instruction in programming Language is called Program.
- An algorithm expressed in a programming language is called a program.
- Programs must obey the grammar of the programming language exactly.
- There is a translator which translates the program into instructions executable by the computer
- Ex. C , C++ , Python etc...




3.Define Pseudo code

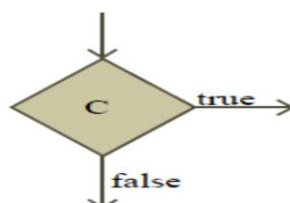
- Pseudo code is a mix of programming language and English.
- It uses the same building blocks as programs, such as variables and control flow.
- pseudo code is not for computers but for human readers.

4.What is a Flowchart

- Flowchart is a diagrammatic notation for representing algorithms.
- They show the control flow of algorithms using diagrams in a visual manner.
- A flowchart is a collection of boxes containing statements and conditions
- which are connected by arrows showing the order of execution

5.List the symbols used in flowchart.

- **Start / End** - are used to indicate the start and the end of an execution : 
- **Parallelogram** boxes represent **inputs/output** 
- **Rectangular** boxes used to indicate a **process** 
- **Diamond** shaped box represent a **condition** with **two** outgoing **arrows**, labeled **true** and **false**.



6.List out the disadvantage of Flowchart.

- To represent algorithms, Flowcharts are less compact than in programming language or pseudo code.
- They cannot display clearly about the basic hierarchical structure of the algorithms.
- Flowcharts do not show disciplined control flow structures

7.Distinguish between a condition and a Statement.

- A statement is a phrase that **commands** the computer to do an action.

Condition.

- A condition is a phrase that describes a **test of the state**.

8.Distinguish between an algorithm and a Program.

Algorithms.

- An algorithm is a **step-by-step** sequence of instruction to solve a problem.
- When an algorithm executed with **input data**, it generates a **process** and ends with **output data**.

Program

- A programming language is a **notation** for expressing **algorithms**
- **Set of Instruction** in programming Language is called **Program**.
- An **algorithm** expressed in a **programming language** is called a program.
- Programs must obey the **grammar** of the programming language **exactly**.
- There is a translator which **translates** the **program** into **instructions** executable by the computer
- Ex. C , C++ , Python etc...

9.What are the types of control flow?

There are three types, they are ,

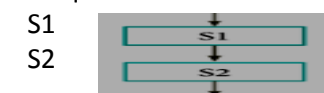
- Sequential
- Alternative
- Iterative

10.Write about sequential control flow with flowchart.

- Statements are executed one after another in the same order

Let **S1** and **S2** be statements.

A sequential statement of S1 and S2 is ..



11. Write about Alternative control flow with flowchart

A condition of the state is tested,

- if the condition is **true**, one statement is executed;
- if the condition is **false**, an alternative statement is executed.

Let **S1** and **S2** – Statements

C – Condition

If **C**

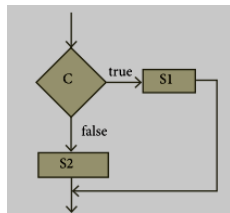
S1

else

S2

1. Test whether **C** is true or false.

2. If **C** is true, then do **S1**; otherwise do **S2**.

**12. Explain Conditional Statement with Flowchart. OR**

Draw a flowchart for conditional statement.

- Conditional statement is used to execute a statement only if a condition is true
- Do nothing if the condition is false.

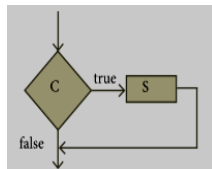
Let **S** – Statements **C** – Condition

if **C**

S

1. Test whether **C** is true or false.

2. If **C** is true then do **S**; otherwise do nothing.

**13. Both conditional statement and iterative statement have a condition and a statement. How do they differ?**Conditional statement

- Conditional statement is used to execute a statement only if a condition is true
- Do nothing if the condition is false.

Iterative Statement

An iterative process executes the same action repeatedly till condition False

14. What is case analysis?

Write about Case analysis control flow with flowchart.

- Case analysis is a multiple branching statement,
- Based on a condition, the control is transferred to one of the many possible points.

If **C1, C2, C3** are conditions

S1, S2, S3, S4 are Statements

case **C1**

S1

case **C2**

S2

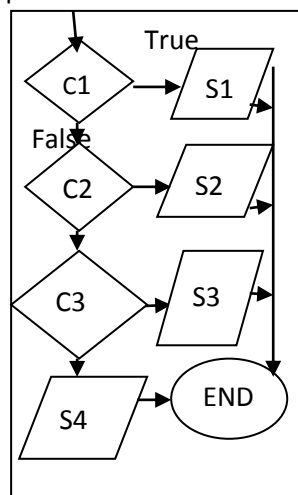
case **C3**

S3

else

S4

- For each case, the problem is solved independently.



- If case condition is **TRUE**, the corresponding statement is executed.
- If none of case is **TRUE** then the default case **S4** is executed.

The cases are exhaustive : At least one case is **TRUE**

The cases are disjoint : The case analysis always execute one case.

15. Write an algorithm that compares two numbers and produce the result as

1. Compare (**a, b**)

2. Case **a < b**

3. Result = -1

4. Case **a = b**

5. Result = 0

6. Else – **a > b**

7. Result = 1

$$\text{compare}(a, b) = \begin{cases} 1 & \text{if } a < b \\ 0 & \text{if } a = b \\ 1 & \text{if } a > b \end{cases}$$

16. Write about Iterative control flow with flowchart

An iterative process executes the same action repeatedly till condition False.

The iterative statement is commonly known as a loop

If **C** is a condition and **S** is a statement, then

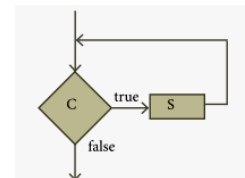
while C

S

This describes the following action:

1. Test whether **C** is true or false.

2. If **C** is true, then do **S** and go back to step 1; otherwise do nothing.

**17. Why is function an abstraction?**

Function is an abstraction of an algorithm,

- The parts of an algorithm are known as functions.
- A function is like a sub algorithm.
- Construct each part (function) separately,
- and then integrate the parts to the complete algorithm.
- A Function is specified by the input property and desired input – output relation.

18. How do we refine a statement? OR

Define Refinement.

After decomposing a problem into smaller sub problems,

- Each sub problem can be expanded into more detailed steps.
- Each step can be further expanded.
- This is known as refinement.

19. Consider the problem of testing whether a triangle is right-angled, given its three sides a , b , c , where c is the longest side. The triangle is right-angled, if $c^2 = a^2 + b^2$

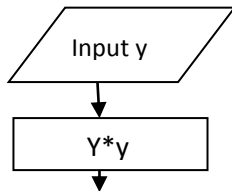
Here a function **square()** is specified as

Square(y)

--input : y

-- output : y^2 ($y*y$)

Flowchart for square()



Algorithm

Right_angled(a, b, c)

--input: $c \geq a, c \geq b$

--output: if $c^2 = a^2 + b^2$ is true result = TRUE otherwise result = FALSE

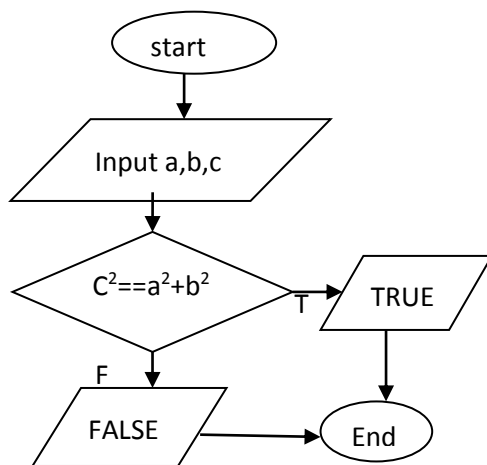
if (square(c) == square(a) + square(b))

result = TRUE it is right angle triangle

else

result = FALSE it is not a right angle triangle

Flowchart



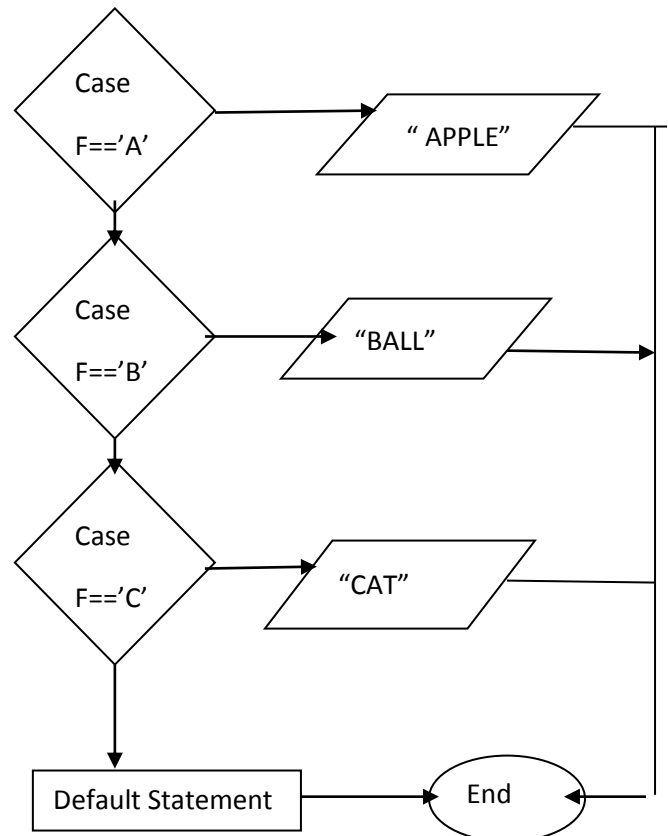
If C is false in line 2, trace the control flow in this algorithm.

- 1 S1
- 2 -- C is false
- 3 if C
- 4 S2
- 5 else
- 6 S3
- 7 S4

If test condition C is FALSE

S3, S4 statements are executed.

20. Draw a flowchart for -3case analysis using alternative statements.



21. Define a function to double a number in two different ways: (1) $n + n$, (2) $2 \times n$

(1) $n + n$

Double 1(n)

-- input(n)

--output $y = n + n$

2) $2 \times n$

Double2(n)

--input(n)

--output $y = 2 \times n$

22. Exchange the contents: Given two glasses marked A and B. Glass A is full of apple drink and glass B is full of grape drink. Write the specification for exchanging the contents of glasses A and B, and write a sequence of assignments to satisfy the specification

Exchange(a, b)

--input: A, B are integers

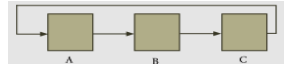
--output: A, B are integers $T = 0$

T := A

A := B

B := T

23.Circulate the contents: Write the specification and construct an algorithm to circulate the contents of the variables A, B and C as shown below: The arrows indicate that B gets the value of A, C gets the value of B and A gets the value of C.



Algorithm

Contents(a,b,c)

--input : a=10,b=20,c=30

--output: a=c ,b=a ,c=b

a=30,b=10 , c=20

Algorithm

Start

Read a,b,c

t=b

b=a

a=c

c=t

print a,b,c

End

24.Trace the step-by-step execution of the algorithm for factorial(4).

factorial(n)

-- inputs : n is an integer , $n \geq 0$

-- outputs : $f = n!$

f, i := 1, 1

while $i \leq n$

f, i := $f \times i$, $i+1$

Algorithm

Read n=4

f=1,i=1

check $i \leq n$ if true execute the following loop

{

f=f*i

i=i+1

}

If false comes out of the loop and

Print f

Output: 24

