COMPUTER SYSTEM IN PROBLEM SOLVING

ELIAS JOSEPH NYAKARUGA

July 20, 2025

Problem solving is the process of identifying a problem, developing a set of procedures (steps) to solve the identified problem and implementing the proposed steps to create a computer program.

Elias Nyakaruga

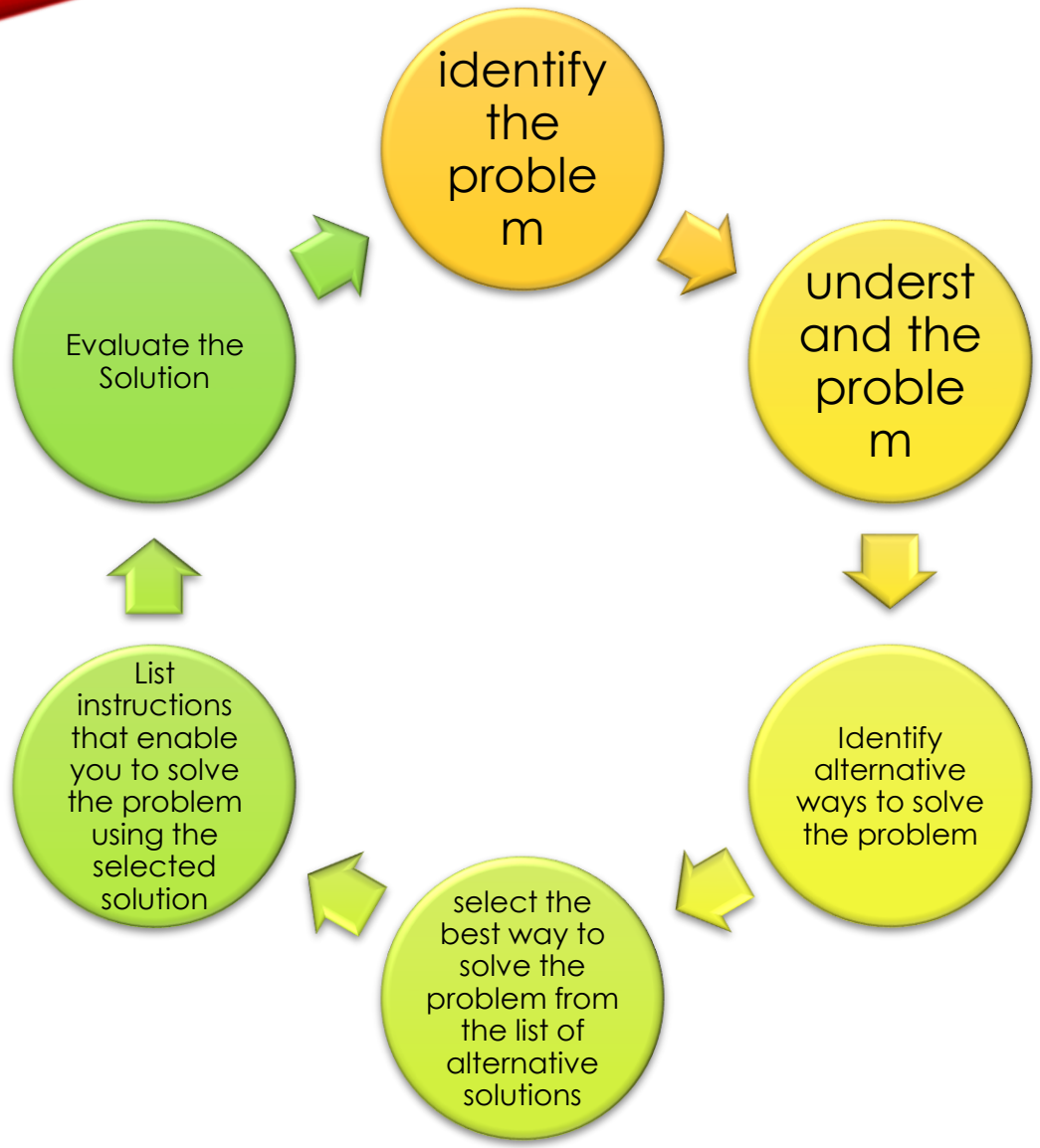
STEPS OF PROBLEM SOLVING

Good problem solving skills can save a lot of time and resources.

There are six steps to solve both algorithmic and heuristics problems

Elias Nyakaruga

Elias Nyakaruga



Step 1: identify the problem

This step involves identifying the problem before starting solving it.

It involves recognizing that a problem exists and understanding what it is.

Elias Nyakaruga

Step 2: understand the problem

This involves understanding the problem in detail, including it causes, effects and potential solutions.

Elias Nyakaruga

Step 3: Identify alternative ways to solve the problem

Elias Nyakaruga

Steps 4: select the best way to solve the problem from the list of alternative solutions

Elias Nyakaruga

Step 5: List instructions that enable you to solve the problem using the selected solution.

1. Break the problem into sub problems to allow for a more organized and focused approach to problem solving
2. Prepare instructions for solving the problem.

Elias Nyakaruga

Step 6: Evaluate the solution

To evaluate or test a solution means to check its result to see if it is correct.

It means to see if the solution satisfies the needs of the people who experience that problem

Elias Nyakaruga

ALGORITHMS

An algorithm is a set of steps leading to the solution of a given problem.

OR

An algorithm is step- by -step procedure or set of instructions designed to solve a specific problem a particular task.

It is method of representing the systematic and logical procedure for solving a problem.

Elias Nyakaruga

IMPORTANCE OF AN ALGORITHM

1. It provides a clear, logical procedure to find solution
2. Break large problems into smaller parts.
3. Optimize time and resource use.
4. Allow evaluation of different solutions.
5. Encourage logical and structured thinking
6. Ensure consistent and correct results.

Elias Nyakaruga

CHARACTERISTICS OF A SUITABLE ALGORITHM

1. Well defined input
2. Well defined output
3. Unambiguity
4. Effectiveness
5. Programming language independence
6. Finiteness
7. Terminator

Elias Nyakaruga

QUALITIES OF A GOOD ALGORITHM

1. Memory utilization
2. Efficiency
3. Cost effective
4. accuracy

Elias Nyakaruga

ALGORITHM DESIGN PRINCIPLES

1. Correctness
2. Efficiency
3. Input and output
4. determinism

Elias Nyakaruga

5. Finiteness
6. Scalability
7. modularity

Elias Nyakaruga

CONSTRUCTION OF ALGORITHM

1. Activities are performed using
   a)defined steps ,
   b)making some decisions,
   c)selecting items,
   d)arranging thing in a given order

Elias Nyakaruga

EXAMPLE

Algorithm to make cup of tea as follows:-

a) Get a teabag, cup, kettle, water, sugar and teaspoon,
b) Put the teabag in the cup,
c) Put some water in the kettle,
d) Boil the water in the kettle,
e) Pour some boiled water into the cup,
f) Add a teaspoon of sugar to the cup,
g) Stir the tea using teaspoon until the sugar is dissolved.

Elias Nyakaruga

STEPS TO CONSTRUCT AN ALGORITHM

Elias Nyakaruga

1. Problem definition

Clearly understand the problem to be solved

To Identify:-

1. Input: What data is given
2. Outputs: what result do we want?
3. Constraints : what rules or conditions must be followed

2. Analyse the problem
3. Decide on the approach for solving problem
4. Review the approach and better alternatives
5. Develop the entire organization of the algorithm
6. Refine the algorithm

Elias Nyakaruga

REPRESENTATION OF ALGORITHM

1. Using flowchart

A flowchart is a diagrammatic representation of a algorithm.

Flowchart is constructed by using different types of symbols each representing a particular step or action taken in an algorithm.

Elias Nyakaruga

ADVANTAGES OF FLOWCHARTS

1. Practical analysis
2. Efficient coding
3. Proper documentation

Elias Nyakaruga

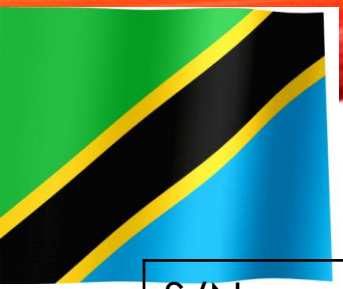
DISADVANTAGES OF FLOWCHARTS

1. Alterations and modifications
2. Loss of details

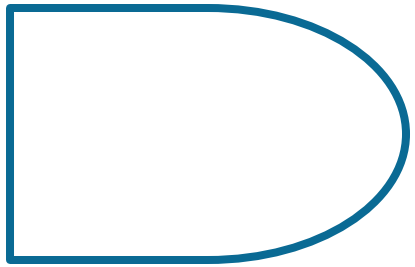
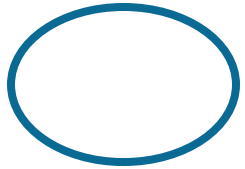
Elias Nyakaruga

Elias Nyakaruga

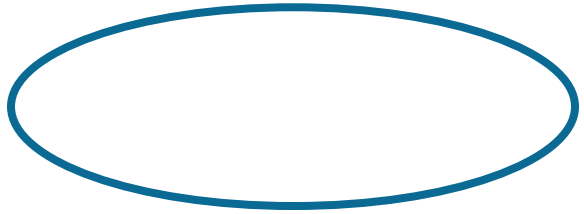
S/N	SYMBOL	DESCRIPTION
1		Process Symbol
2		Flowlines Symbol
3		Decision Symbol



Elias Nyakaruga

S/N	SYMBOL	DESCRIPTION
4		Delay or wail symbol
5		Connector Symbol:
6		Input and Output Symbol

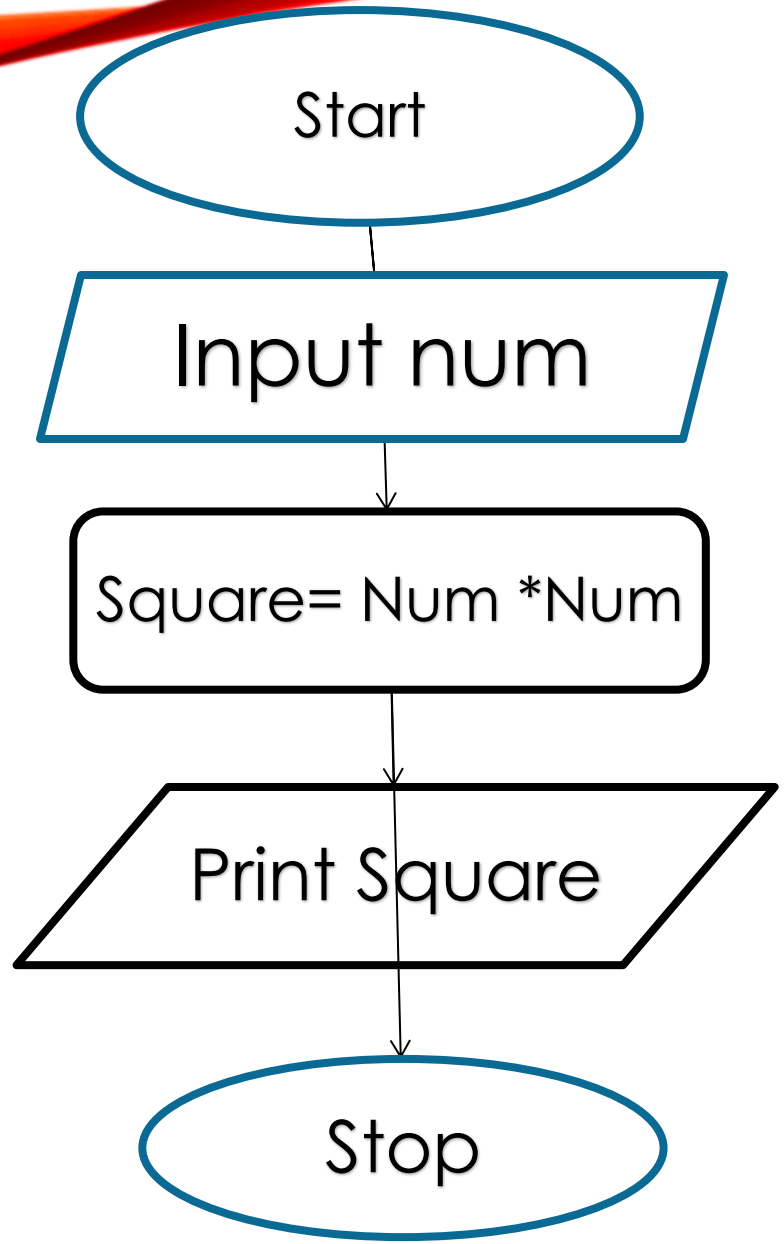
Elias Nyakaruga

S/N	SYMBOL	DESCRIPTION
7		Printed Document specifies
8	 	Terminal symbols:

STEPS FOR DRAWING A FLOWCHART

1. Know the purpose of your flowchart
2. Identify steps
3. Collect information
4. Add shapes and symbols
5. Determine the flow
6. Add the decision points
7. Include start and end points
8. Add connectors

Elias Nyakaruga



Elias Nyakaruga

2. Using pseudocode

Pseudocode is the textual representation of an algorithm containing a detailed description of instructions.

Pseudocode is a non formal language that helps programmers to write algorithms.

There are no accurate formatting or syntax rules for writing pseudocodes.

Elias Nyakaruga

ADVANTAGE OF PSEUDOCODE

1. It can be easily modified unlike flowcharts.
2. It can be read and understood easily.
3. It is easier to convert the pseudocode to a programming language than flowchart.
4. It is helpful in structured design.
5. It can be easily written using any word processor or text editor.

Elias Nyakaruga

DISADVANTAGES OF PSEUDOCODE

Elias Nyakaruga

1. It is more difficult for beginner to follow the logic or write pseudocode than flowchart
2. There is no standard way of writing pseudocode

Elias Nyakaruga

In pseudocode keywords are used to indicate common input, processing operations and output.

They can be written entirely in uppercase or start with capital letter.

KEYWORDS USED IN PSEUDOCODE

item	Keyword	Description
1	START or BEGIN	This is used whenever you start your pseudocode
2	INPUT, READ, WRITE or GET	These are keywords used for inputting data
3	PRINT, DISPLAY, SHOW	This will show your output to a screen or any output device
4	COMPUTE, CALCULATE or DETERMINE	This is used to calculate the results of an expression
5	SET	it is used to initialize values or assign a value to a variable
6	INCREMENT	It is used when you want to increase the value of a variable by one
7	DECREMENT	This keyword is meant to reduce the value of variable by one
8	END	Used to indicate the endpoint of the program or a process

Elias Nyakaruga

RULES FOR WRITING PSEUDOCODE

1. Only one statement should be written for each line

Elias Nyakaruga

2. The Input, Output and process statements must be clearly stated using standard keywords such

READ, WRITE, PRINT, DISPLAY and RETURN

Example

```
START
    READ a, b, c
    c = a + b
    PRINT c
END
```

Elias Nyakaruga

3. Each initial keyword should be capitalized

```
START
  READ base, height, area
  COMPUTE:
  Area = 0.5 * base * height
  PRINT area
END
```

Elias Nyakaruga

4. Show indentation for the pseudocode.

Elias Nyakaruga

```
BEGIN
    READ Num1, num 2
  If (num 1 < num2)
      SET Min = num1
  Else
      SET Min = num2
    PRINT Min
END
```

5. Pseudocode should show start and end of executable statements and control structures.
6. The statements should be short, clear and readable.

Elias Nyakaruga

EXAMPLE

1. The Pseudocode for calculating the sum and average will be as follows:

```
START
    PRINT "Enter two Numbers."
    INPUT X,Y
    SUM = X+Y
    AVERAGE =  SUM/2
    PRINT SUM
    PRINT AVERAGE
END
```

Elias Nyakaruga

COMPARISON BETWEEN FLOWCHART AND PSEUDOCODE

SN	CRETERIA	PSEUDOCODE	FLOWCHART
1	Representation	Textual representation of an algorithm	Graphical representation of an algorithm
2	Level of Details	It is flexible to accommodate many details	If many details are needed the flowchart becomes more complex
3	Alteration	The pseudocode is flexible to alteration after it has been designed	It is difficult to modify once it has already been designed
4	Format	There are no specific symbol to use, only textual description	Uses specific symbols to present the algorithm

Elias Nyakaruga

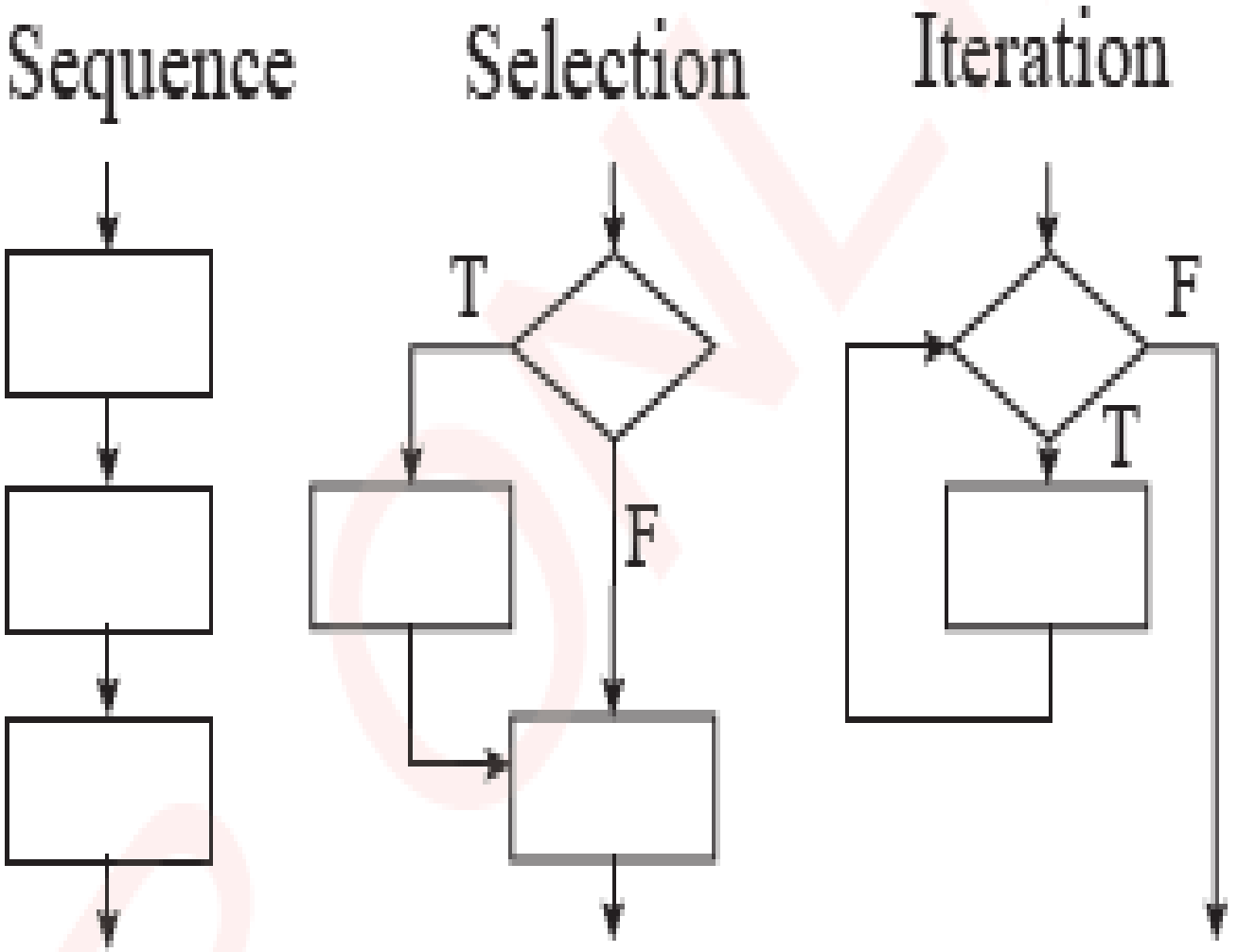
CONTROL STRUCTURES IN FLOWCHARTS AND PSEUDOCODE

Control Structures can be used to represent control of the process flow in terms of:-

a) Sequence
b) Selection
c) Iteration

Elias Nyakaruga

Elias Nyakaruga



1. Sequence control structure in flowcharts

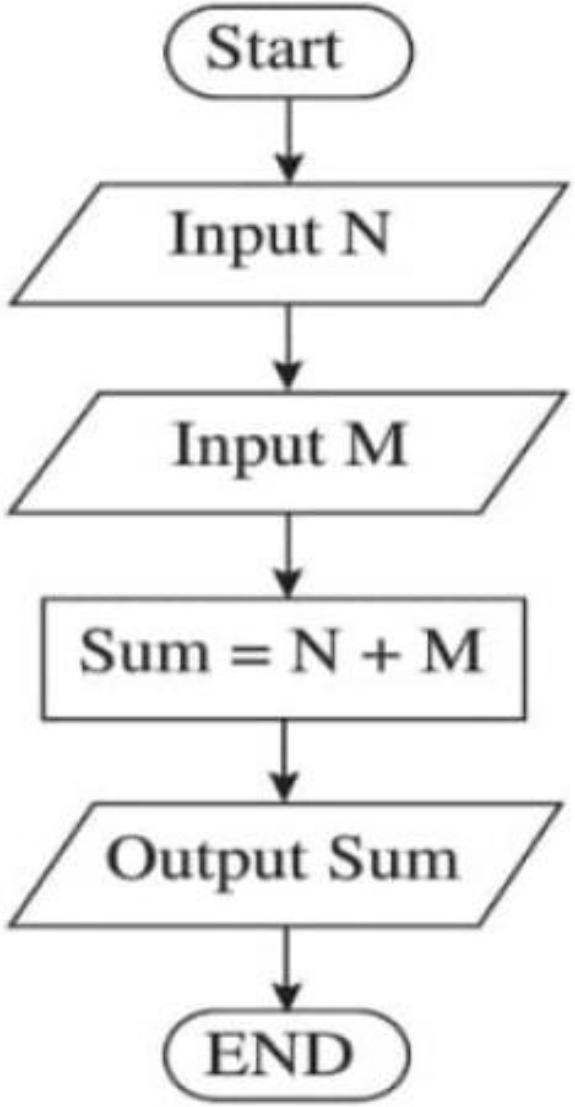
The computer reads an instruction from a program file line by line sequentially, starting from the first line.

Then executes the tasks that are arranged sequentially.

Elias Nyakaruga

1. Example of flowchart using sequential control structure for algorithm to add two numbers.

Elias Nyakaruga



Elias Nyakaruga

2.Selection control structure in flowcharts

The selection control structure is used to decide on a program. The selection control structure makes the best paths between the two or more options given in the logic of the program.

One condition is evaluated first and then flows into one of two or more paths.

Once the conditional execution is finished, it will rejoin before leaving the structure.

It uses IF….THEN or

IF….THEN ….ELSE or

a CASE structure in pseudocodes and program.

Elias Nyakaruga

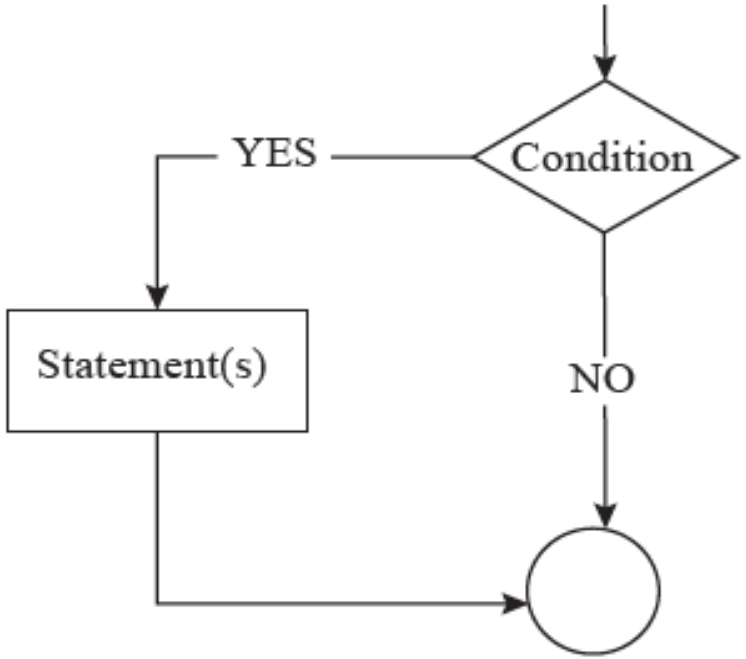
1. The syntax for IF……THEN is

IF *(Condition) THEN*
Statement
END IF;

Elias Nyakaruga

The Corresponding flowchart

Elias Nyakaruga

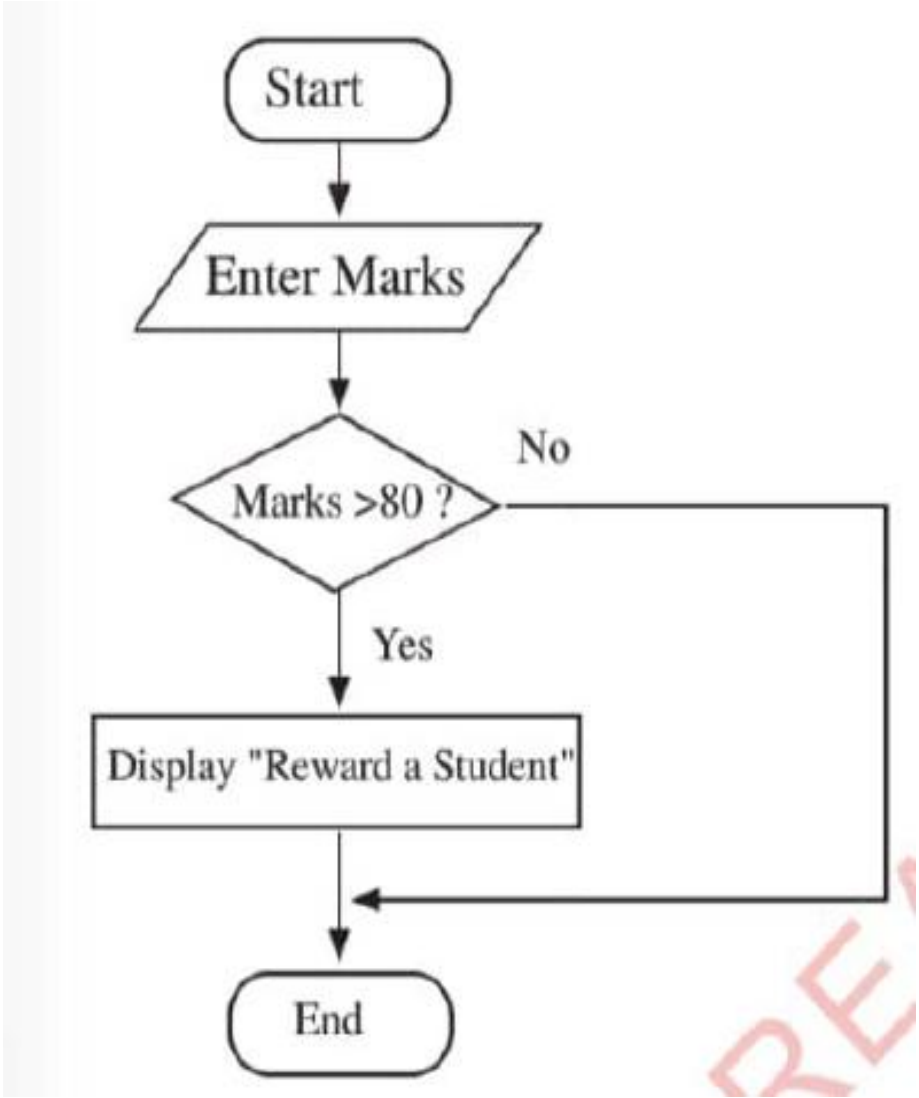


EXAMPLE

IF ….THEN selection control structure

Elias Nyakaruga

Write an algorithm using flowchart for the headmaster of Kingili Secondary school who want attain an average mark above 50%



Elias Nyakaruga

IF …THEN….ELSE

If the condition is true, it executes process 1;

if the condition is false, it will perform the second process (process 2).

In this case, process 1 or process 2 will only be implemented depending on the given condition.

Elias Nyakaruga

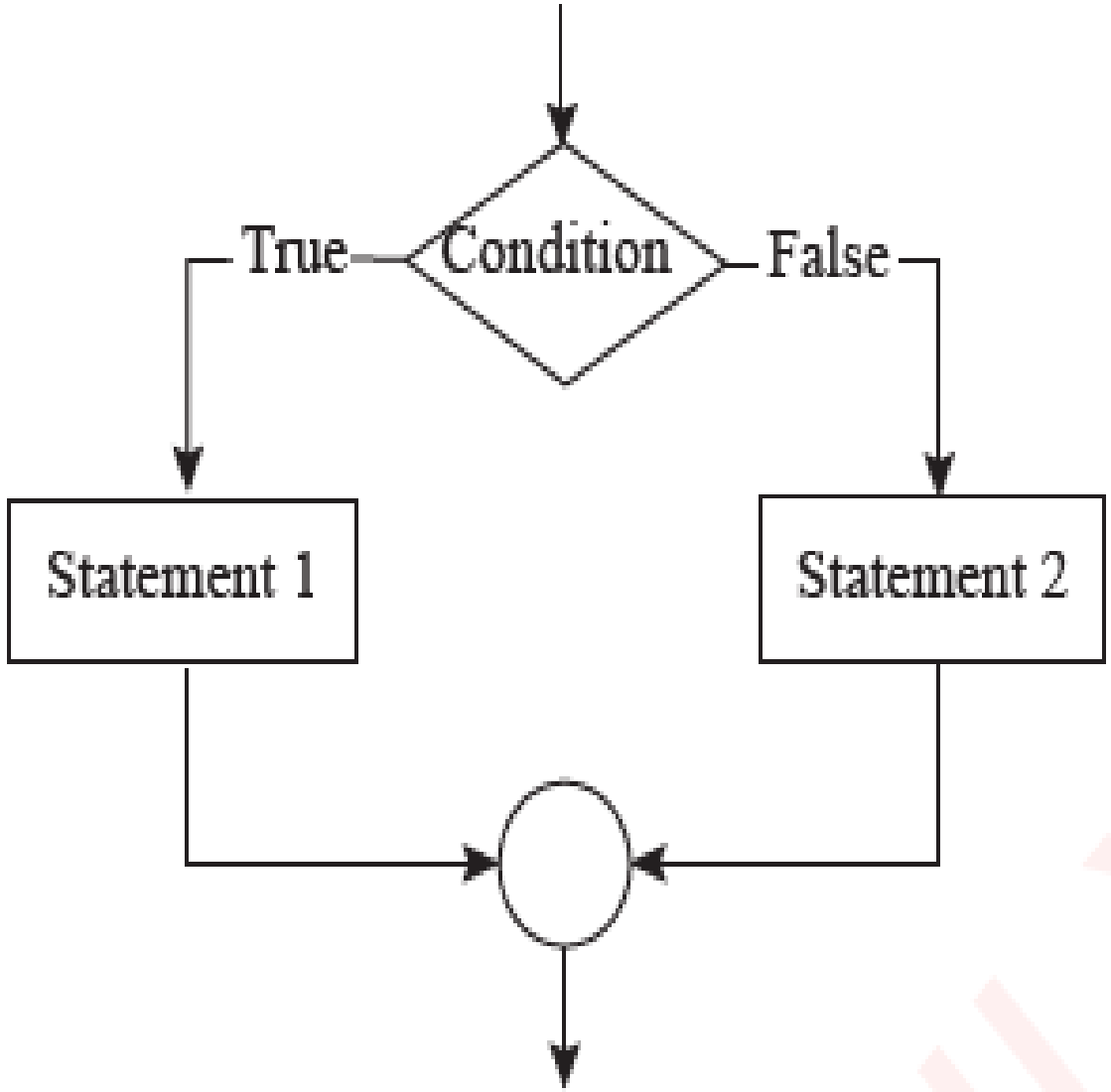
The syntax of If…..THEN…..ELSE is shown as follows:

IF (condition) THEN
Statement 1
ELSE
Statement 2
ENDIF;

Elias Nyakaruga

Example of IF…THEN…ELSE selection

Elias Nyakaruga



Example of IF-THEN-ELSE selection

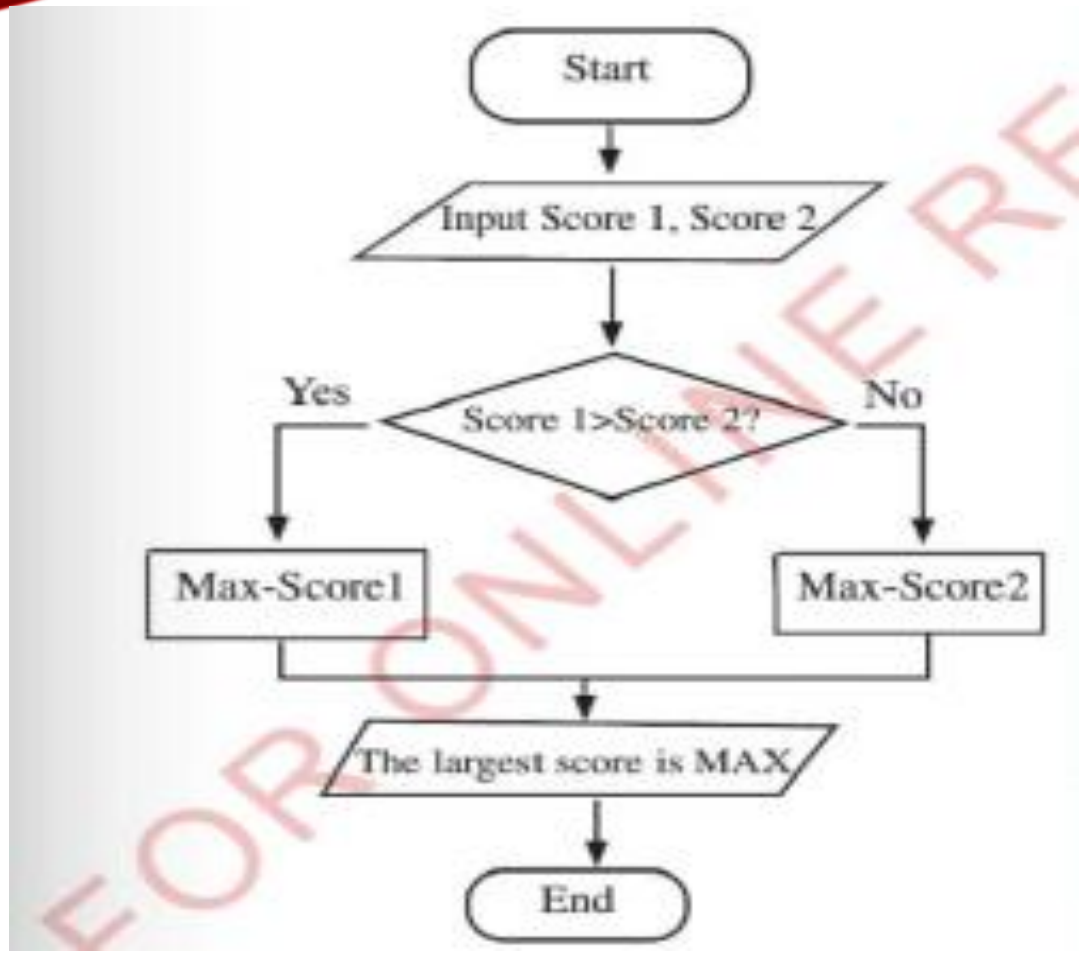
Write an algorithm using a flowchart that reads two scores of a student, determines the largest score and print the largest score with the message that identifies it.

Elias Nyakaruga

ALGORITHM

a. Read the first score and assign it tobe score 1
b. Read the second score and assign it to score2
c. If score 1 is greater than score 2 then display " The largest score is score 1"
d. Else is score 2 is greater than score 1 then display " The largest score is score 2"
e. end

Elias Nyakaruga



Elias Nyakaruga

IF…. THEN …. ELSE IF

Elias Nyakaruga

You can introduce this IF statement if you want to select an action from several alternatives.

This statement introduce the additional conditions.

If ***condition1*** is false the *Else if* executes another condition.

In this form of IF statement more than one *Else if* can be used.

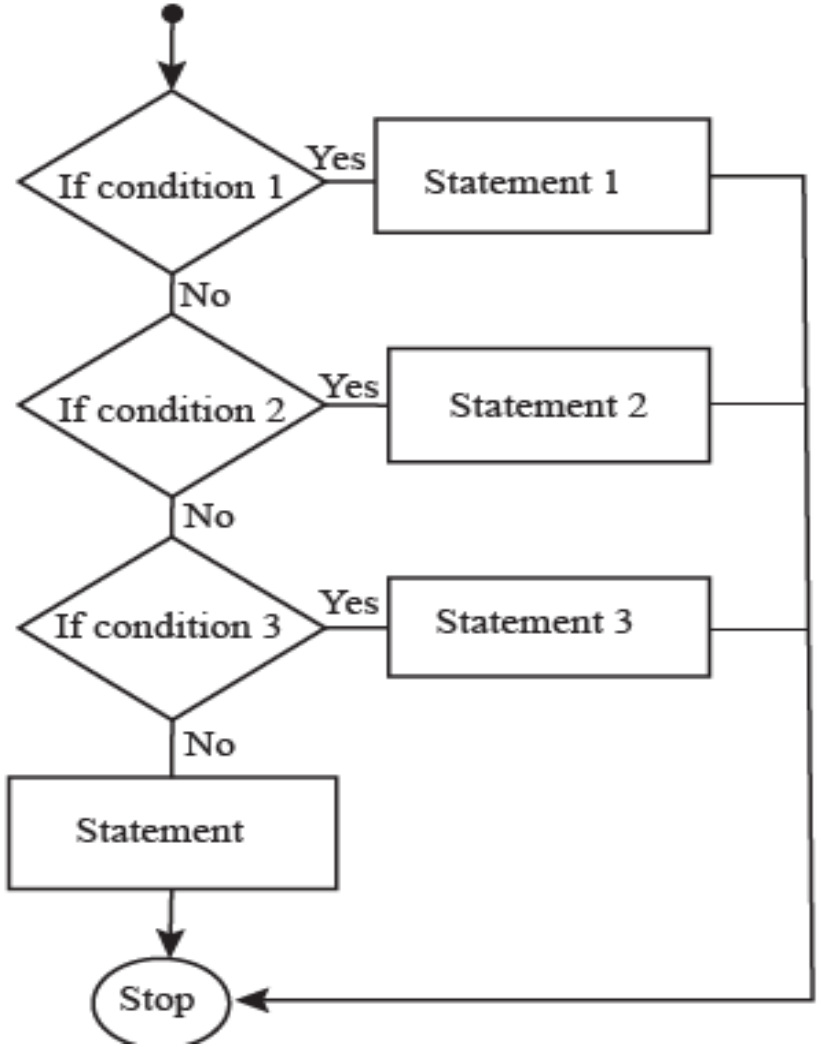
All conditions in the statement are evaluated one after another from top to bottom, when one of them is true then it will be executed.

If all conditions are false, the sequence in *Else* clause is executed.

The syntax for IF…THEN…ELSE IF :

```
IF (condition1) THEN
        sequence_of_statements1
ELSE IF (condition2) THEN
        sequence_of_statements2
ELSE
        sequence_of_statements3
END IF;
```

Elias Nyakaruga



Elias Nyakaruga

EXAMPLE OF IF-THE-ELSEIF
Selection control structure

Write an algorithm to determine the weather conditions of your environment using a flowchart diagram

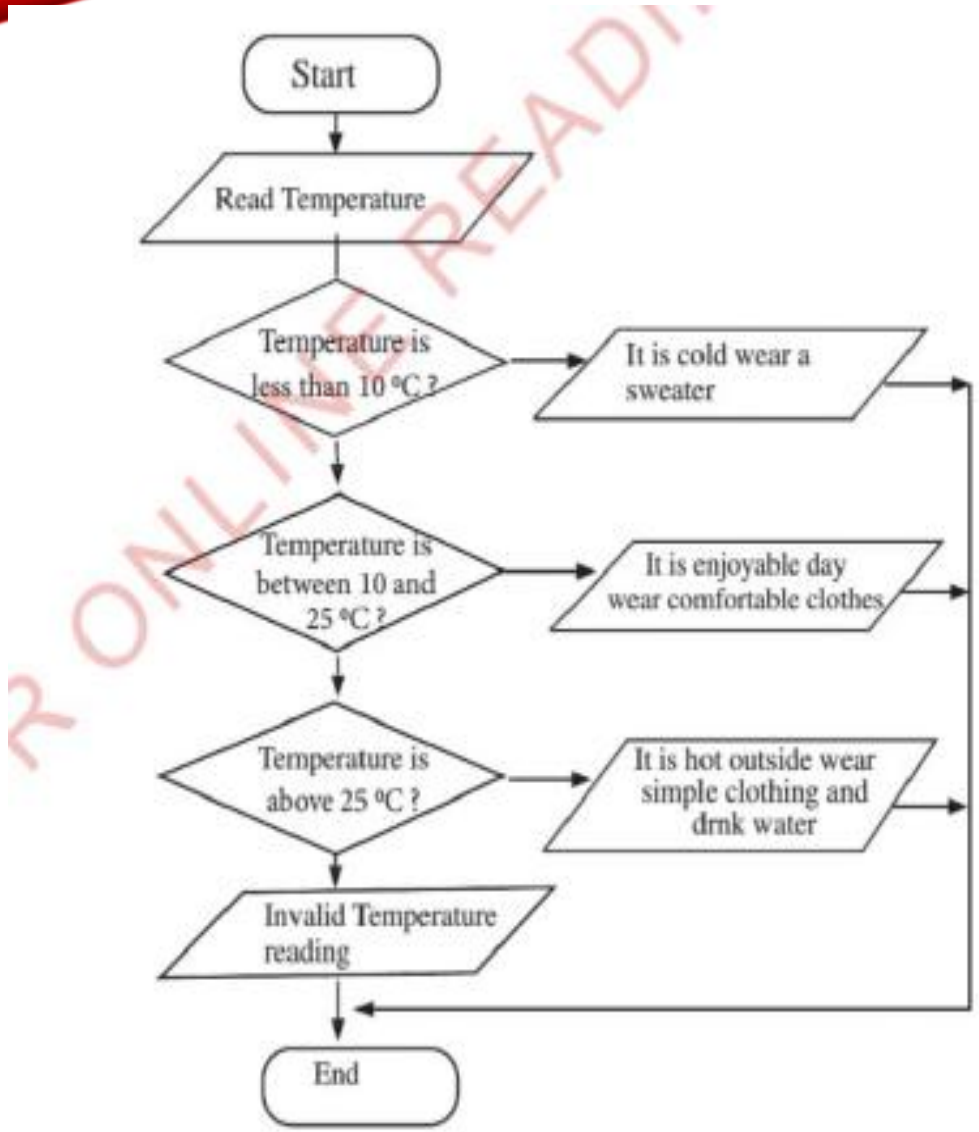
Elias Nyakaruga

ALGORITHM

a) Read the current temperature
b) If the temperature is less than 10 the display “it is cold wear a sweater”
c) Else if the temperature is between 10 and 25 then display “it is enjoyable day wear comfortable clothes”
d) Else if the temperature is above 25 then display “ it eis hot outside wear simple clothing and drink water”
e) Else display “ Invalid temperature reading
f) End

Elias Nyakaruga

Elias Nyakaruga



3.Iteration control structure in flowchart

Iteration also known as looping means repeating steps or instructions over and over.

Iteration is used when there is a need to execute a statement or block of statements several times.

Elias Nyakaruga

EXAMPLES OF LOOPING IN REAL LIFE

1. A cook in the kitchen may try different ingredient or adjust different steps of cooking process slightly until the food tastes nice as required.
2. The carpenter has released multiple versions of office chair product.
3. When a musician begins with one version of song and improves it by modifying speed or a few notes until the best final product has been produced.

Elias Nyakaruga

4. In agriculture farmers may use irrigation by farmers to water their crops . Continuous watering of crops until they sufficiently hydrated can be simulated using a loop.
5. Teachers often need to process student grades and calculate averages.

Elias Nyakaruga

Elias Nyakaruga

The loops structure consists of two main parts

a) **Loop body** - this represents the statements to be repeated.

b) **Loop control** - this specifies the entry/exit condition of the loop

Elias Nyakaruga

A loop employed when you wish to repeat a set of instructions as long as a certain condition is true.

It consists of condition that is checked before each iteration.

If the condition becomes false, the loop terminates

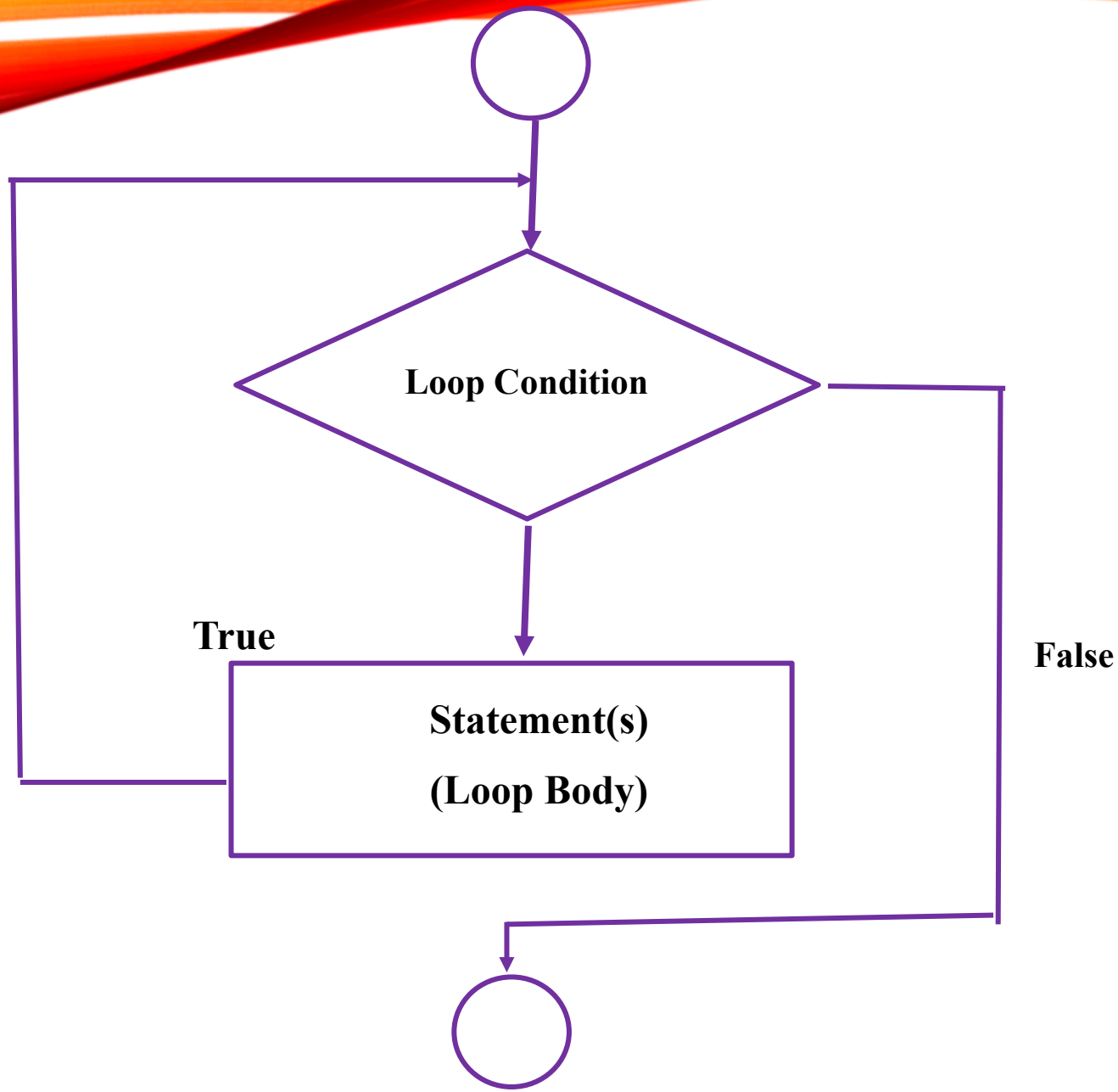
General syntax of a loop

(condtion)

Statement(s) to be executed

Elias Nyakaruga

Elias Nyakaruga



EXAMPLE OF LOOP ITERATION CONTROL STRUCTURE

Consider that you have a "KIBUBU" and that you wish to save money on it. Write an algorithm that asks the user to input coins in Tanzania shillings (Tshs) repeatedly and then add the sum to the "KIBUBU" until exceeds or equals 6000 shillings. Construct this algorithm and draw a corresponding flowchart

Elias Nyakaruga

ALGORITHM

i. Start a loop with the condition that KIBUBU contains less than 6000

ii. Within the loop
   a) Prompt the user to input the amount of coins (in Tshs)
   b) Add the input amount to KIBUBU
   c) Check if KIBUBU is still less than or equal to 6000
   d) If the condition is false exit the loop.

iii. End the loop

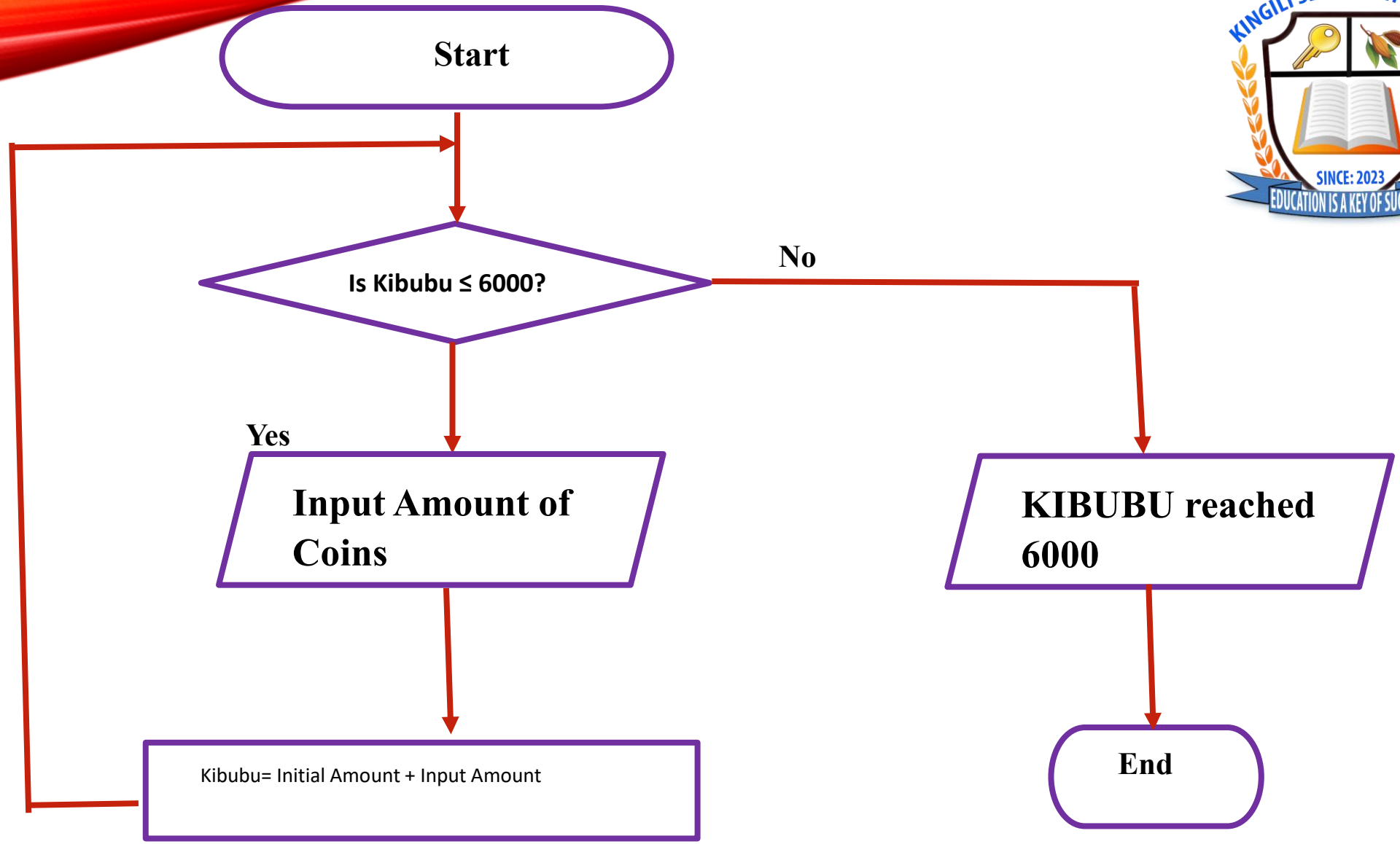
iv. Display a message to the user that their KIBUBU now contains 6000 shillings or more.

v. End

Elias Nyakaruga

FLOWCHART

Elias Nyakaruga



Elias Nyakaruga

The end of Topic