

Q1) Design a flowchart, Pseudocode, Algorithm for processing a customer order at a restaurant, including handling special requests (Like add on).

i. Pseudocode

Start

Print "Enter your order"

Read order

If Customer has Special Requests

Then Handle Special Requests, Confirm Add-ons and Changes, Update Order with Special Requests

Else prepare standard order

End if

Print "Your Order has been Placed"

End

ii. Algorithm

Start

Receive Order: Take the customer's order, including any special requests or add-ons.

Check for Special Requests: If the customer has special requests (add-ons):

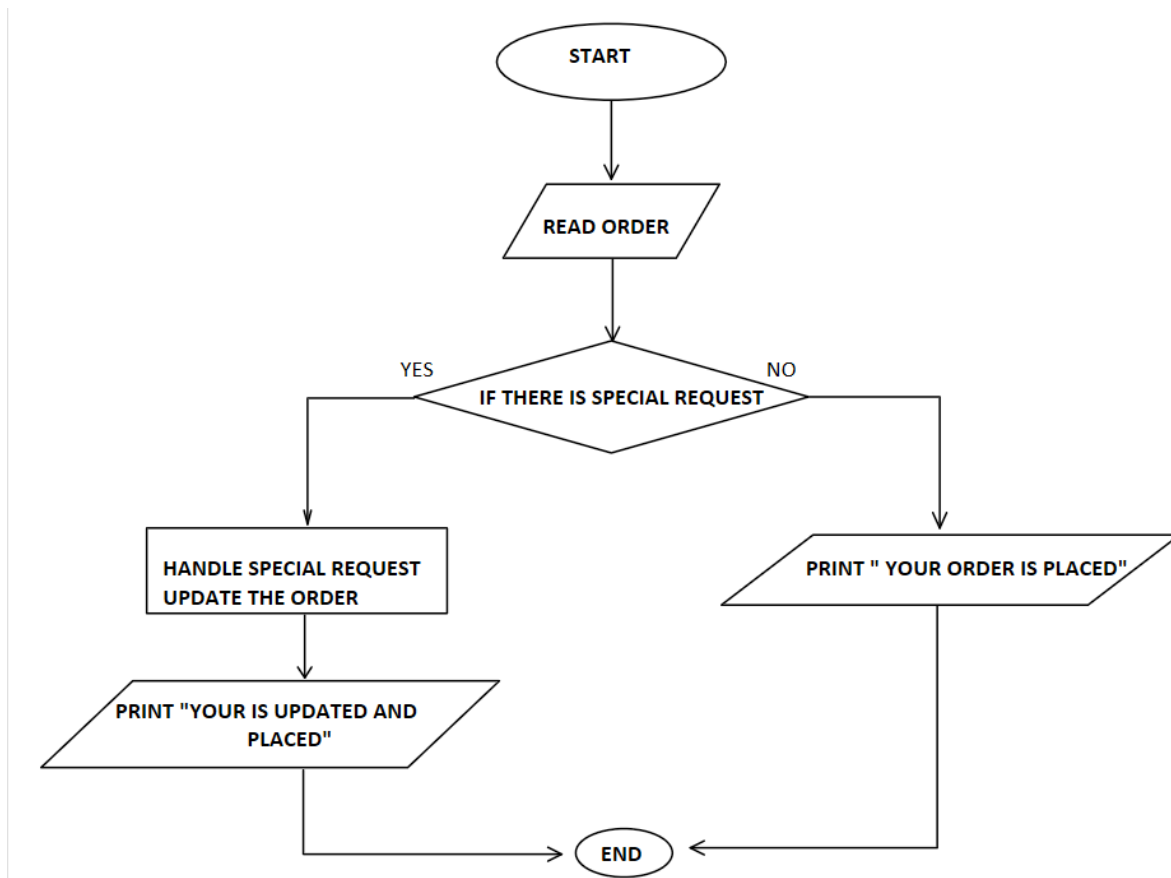
Handle Special Requests: Note any specific instructions for the order.

Confirm Add-ons and Changes: Verify the special requests with the customer to ensure accuracy.

Update Order: Incorporate the special requests into the order details.

If there are no special requests: Prepare Standard Order

End

iii. Flowchart

Q2) Design a flowchart, Pseudocode, Algorithm for handling a customer's deposit transaction at a bank, including checks for account validity and deposit amount conditions.

Solution:

i. Pseudocode

Start

Input Account_Number

Input Deposit_Amount

Set accno=123456789

Set min_deposit= 5000

If Account_Number== accno

Then If Deposit_Amount > min_deposit

Then Print "Transaction Completed"

Else Print "Amount Too Low"

End If

Else Print "Invalid Account number"

End If

End

ii. Algorithm

Start

Obtain account number and deposit amount from the customer.

Check if the provided account number is valid proceed to check the deposited amount.

if the account number is invalid reject transaction.

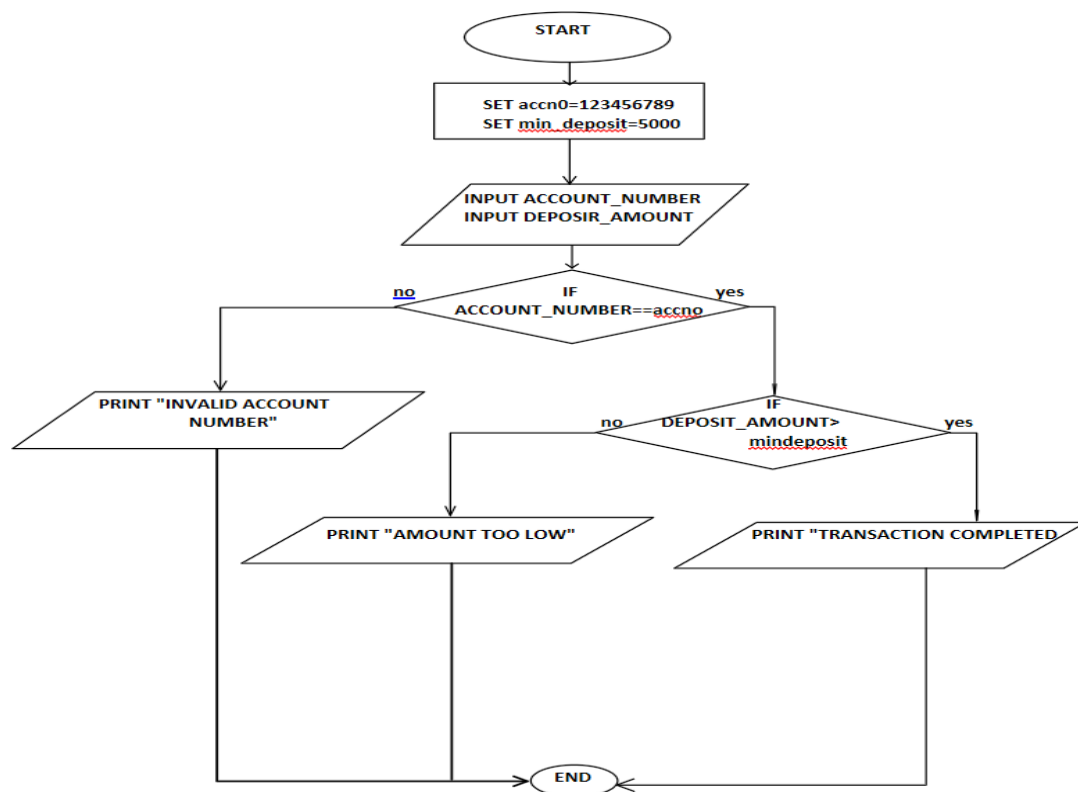
Check If the deposit amount is greater than the minimum required deposit proceed with the deposit.

if it is less than the minimum required deposit than reject transaction.

Notify the customer that the transaction has been completed successfully or rejected.

End

iii. Flowchart



Q3) Design a flowchart, Pseudocode, Algorithm to determine which of three provided numbers is the greatest.

i. Pseudocode:

Start

Print "Enter three numbers to find the largest"

Read a , b , c

If a>b and a>c

then largest = a

Else If b>a and b>c

then largest = b

Else largest = c

End If

Print "The largest number is: ", largest

End

ii. Algorithm:

Start

Take three numbers to find the largest.

Let the three number be a,b and c.

Now check if a is greater than b and a is also greater than c

then declare a is largest number.

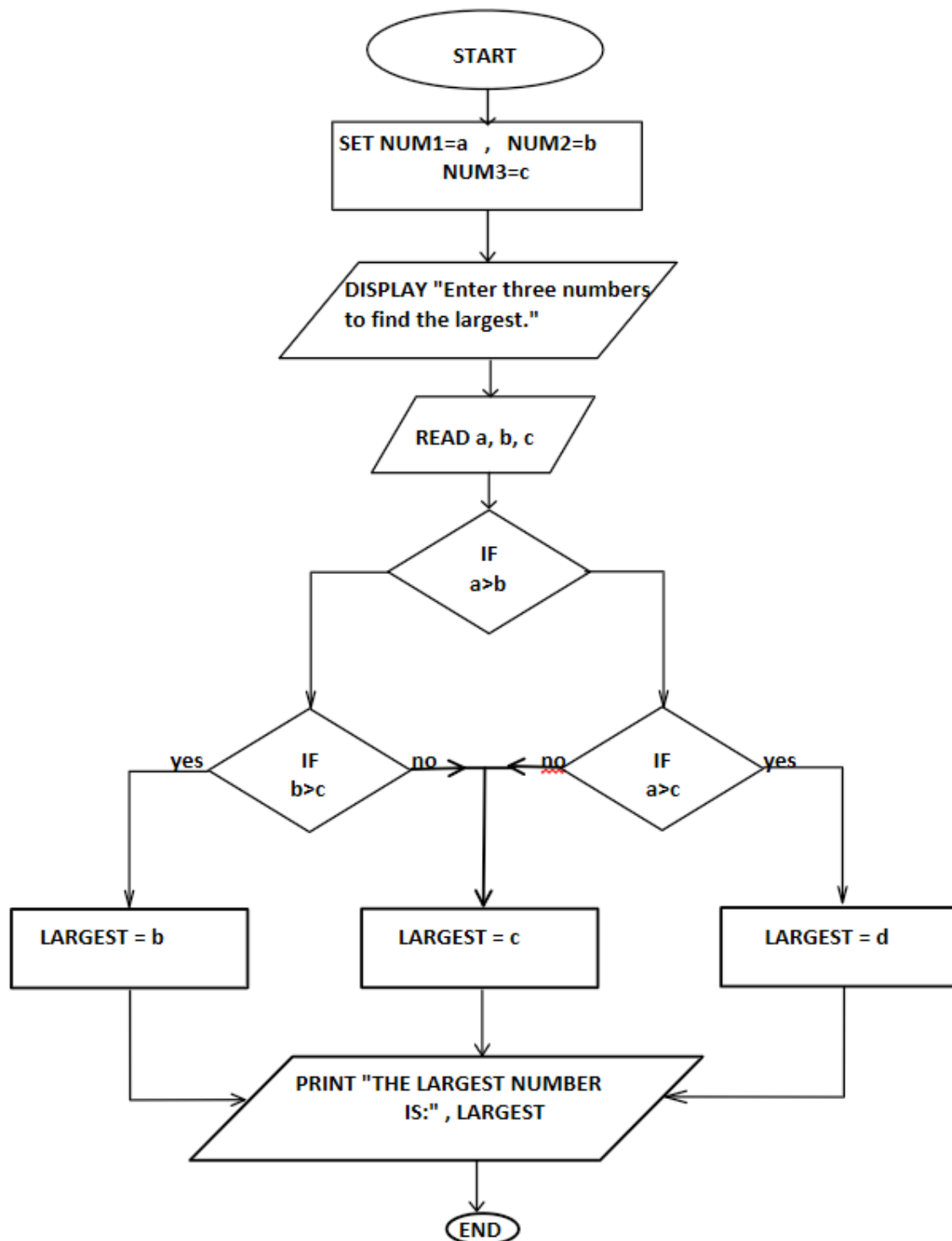
Or check if b is greater than a and b is also greater than c

then declare b is largest number.

Or declare c is the largest number

End

iii. Flowchart



Q4)Implement an algorithm where the user enters a number, and an appropriate month is displayed.

Algorithm:

Start

Let the '1'= january

'2'= february

'3'=march

'4'=april

'5'=may

'6'=june

'7'=july

'8'=august

'9'=september

'10'=october

'11'=november

'12'=december

Ask user to enter a number between 1 and 12

Take number

Now check if the number is equal to 1

then tell user that the month is january

Or check if the number is equal to 2

then tell user that the month is february

Or check if the number is equal to 3

then tell user that the month is march

Or check if the number is equal to 4

then tell user that the month is april

Or check if the number is equal to 5

then tell user that the month is may

Or check if the number is equal to 6

then tell user that the month is june

Or check if the number is equal to 7
 then tell user that the month is july

Or check if the number is equal to 8
 then tell user that the month is august

Or check if the number is equal to 9
 then tell user that the month is september

Or check if the number is equal to 10
 then tell user that the month is october

Or check if the number is equal to 11
 then tell user that the month is november

Or check if the number is equal to 12
 then tell user that the month is december

Or tell user to enter number only between 1 and 12

End

**Q5) Create pseudocode a small calculator which only does '+' or '-' Operations.
(Hint: Take three variable inputs with one being used for the operator)**

PSUEDOCODE:

Start

Print "Please enter two number"

Read num1

Read num2

Print "please enter the airthmetice operator '+' or '-' "

Read operator

If operator == '+'
 then Result = num1 + num2

Else If operator == '-'
 then Result = num1 - num2

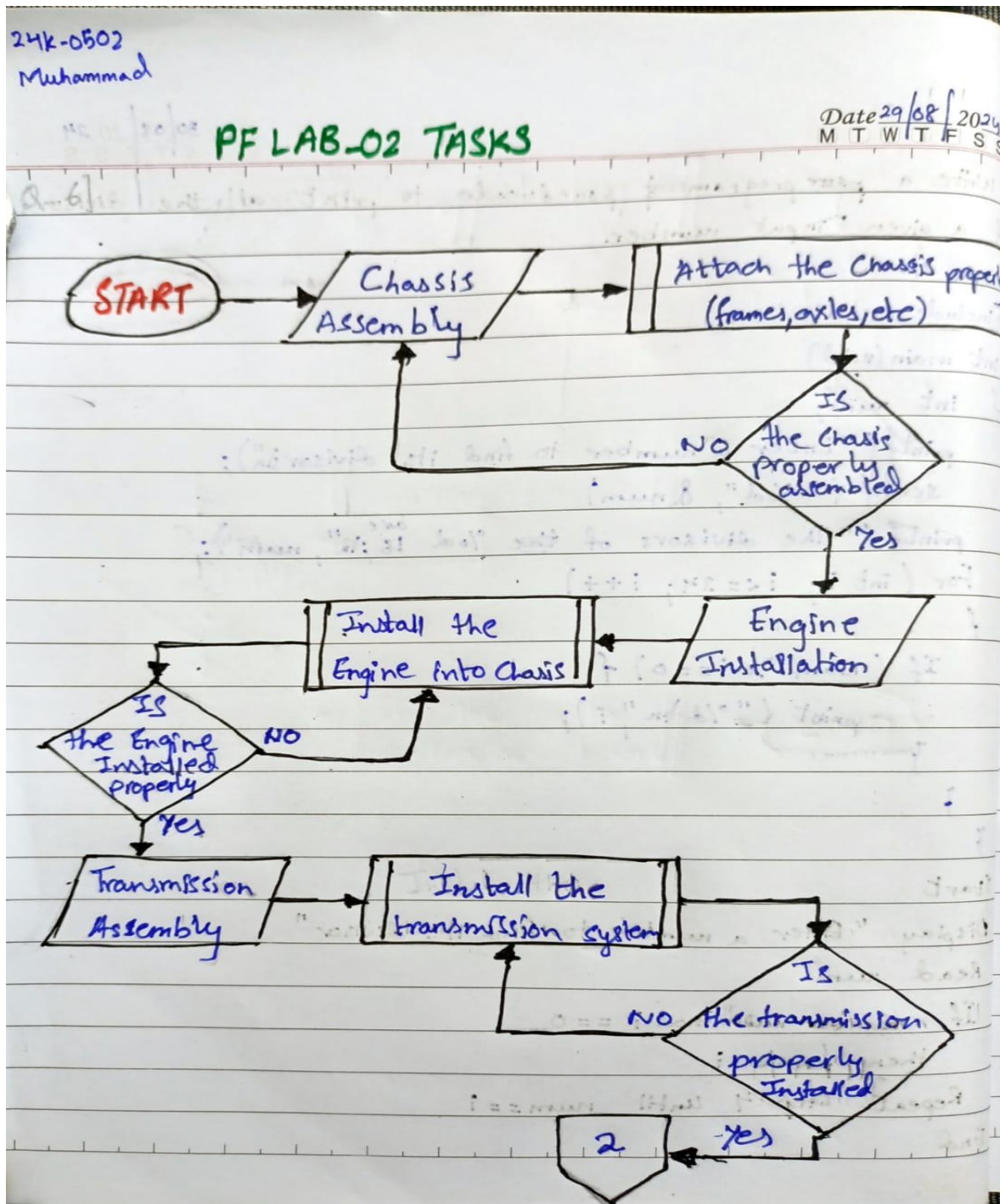
Else Print " Enter correct operator!"

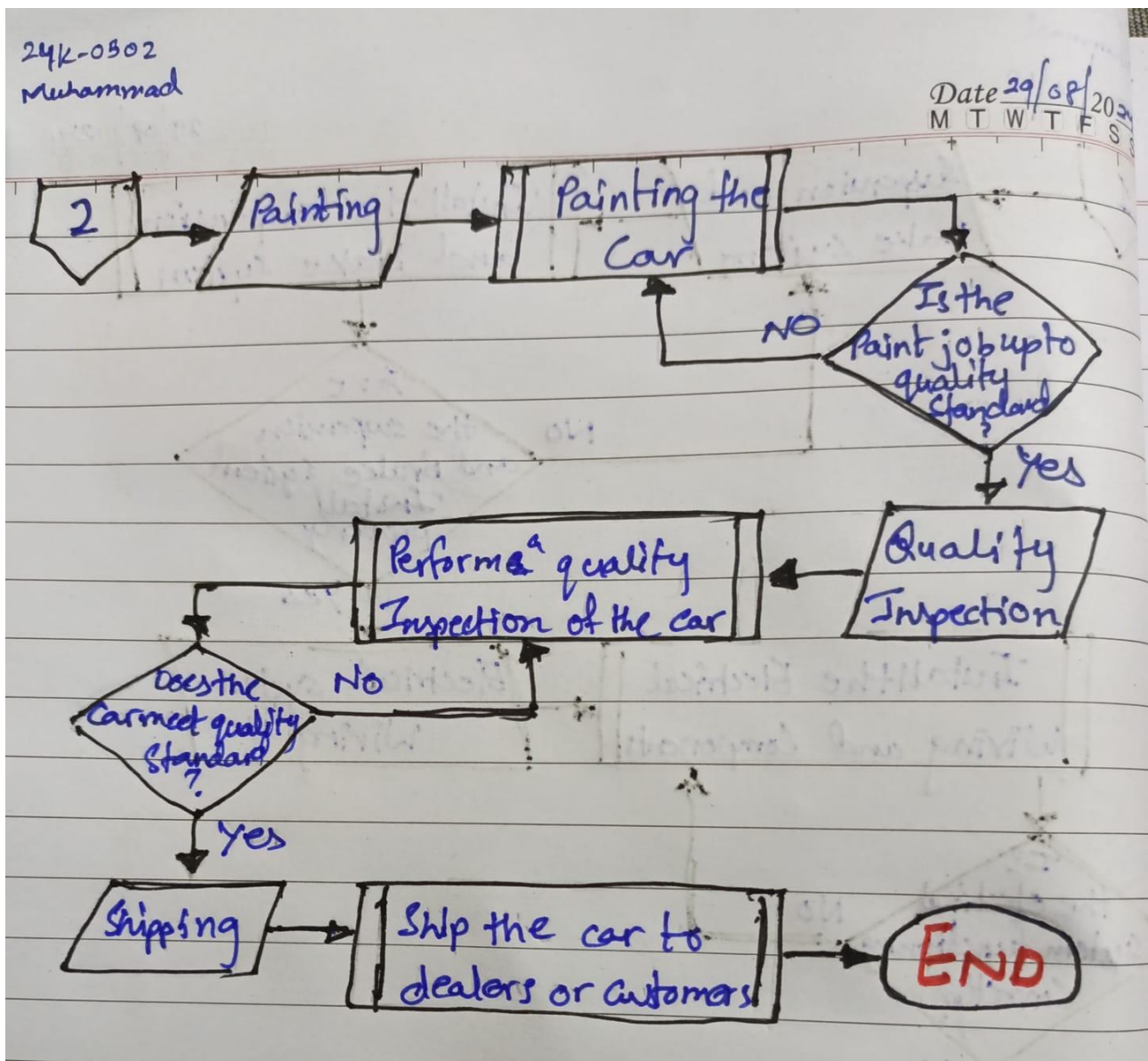
End If

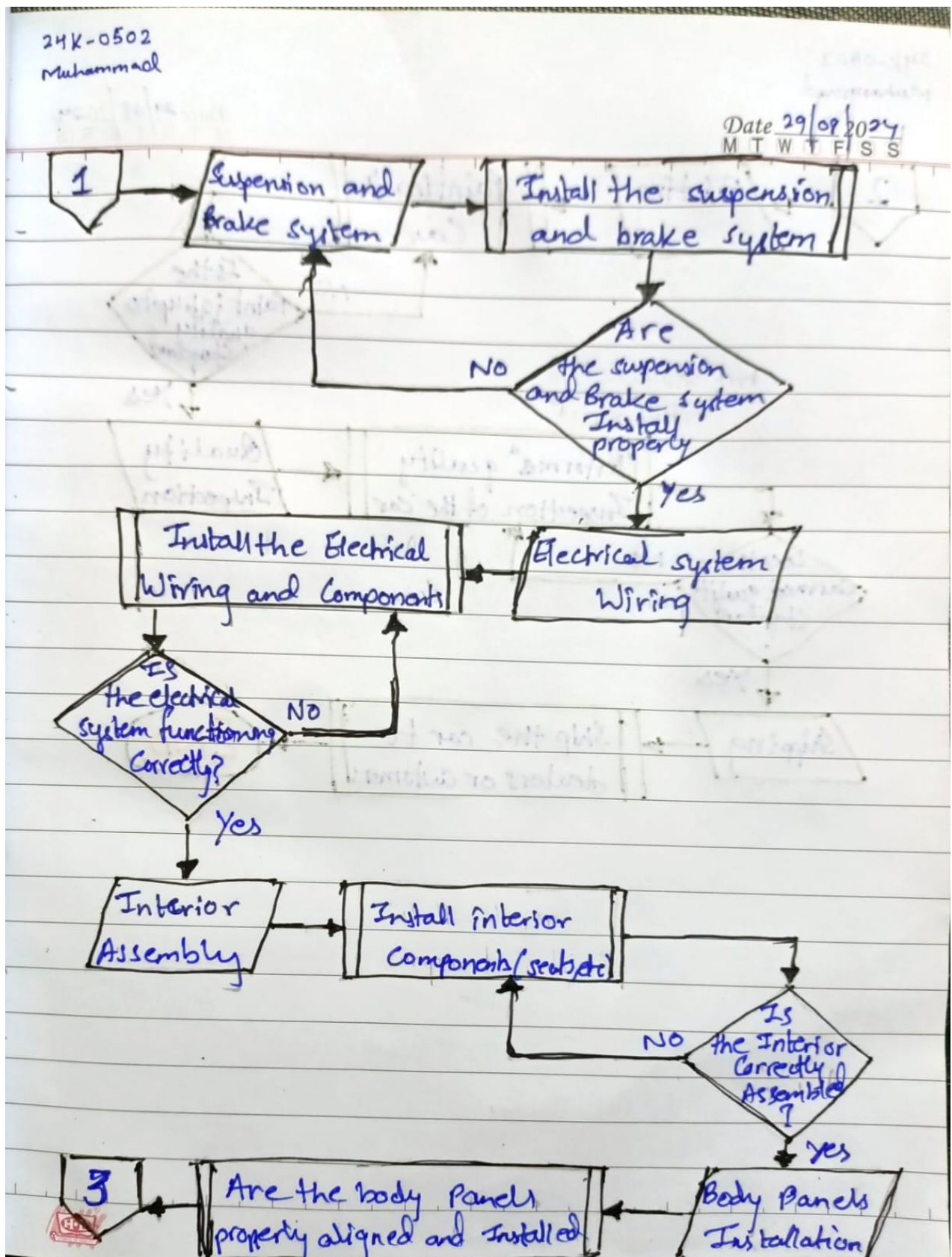
Print "Result"

End

Q6) You are working at Toyota Indus Motors and want to assemble a car.
Design a flowchart with proper process modules and decision structures to replicate a pipeline production







Q7) Implement an algorithm for making a simple calculator with all the operators (+, -, *, /, %)

ALGORITHM:

Start

Ask user to Enter two numbers:

Take num1 and num2 from the user

Now ask user to enter the arithmetic operator

Take operator from the user

Check if operator is equal to '+'

Then result = num1 + num2

Or check if operator is equal to '-'

Then result = num1 - num2

Or check if operator is equal to '*'

Then result = num1 * num2

Or check if operator is equal to '/'

Then result = num1 / num2

Or ask user to enter a correct operator

End if

Give result

End

Q9) Why we use .gitignore?

Ans) A .gitignore file is used in a Git repository to specify which files and directories should be ignored by Git. When you have a .gitignore file, Git will not track or include the listed files in version control, even if they are present in the working directory. This helps to keep the repository clean and manageable. Here are some common reasons why you would use a .gitignore file:

1. Prevent Sensitive Data from Being Committed
2. Exclude Unnecessary Files
3. Improve Performance
4. Avoid Platform-Specific Files
5. Manage Dependencies Separately

Q10) Difference between Algorithm and Pseudocode?

ALGORITHM	PSEUDOCODE
-----------	------------

<ul style="list-style-type: none">• An algorithm is a well-defined, step-by-step procedure or set of rules designed to perform a specific task or solve a particular problem. It is a logical sequence of actions that provides a solution in a finite amount of time.	<ul style="list-style-type: none">• Pseudocode is a high-level description of an algorithm using a mixture of natural language and programming-like syntax. It is not actual code but is designed to represent the logic and flow of an algorithm without the specifics of a particular programming language.
<ul style="list-style-type: none">• An algorithm is the conceptual solution to a problem	<ul style="list-style-type: none">• While pseudocode is a way to express that solution in a readable and structured format.
<ul style="list-style-type: none">• Algorithms are more abstract and focus on the logical steps needed to solve a problem	<ul style="list-style-type: none">• Pseudocode, while still abstract, adds more structure to represent how those steps would translate into code.
<ul style="list-style-type: none">• Algorithms are used to outline the problem-solving approach	<ul style="list-style-type: none">• While pseudocode is often used as an intermediary step between the algorithm and the actual code implementation, helping bridge the gap between abstract thinking and concrete coding.