**FLOWCHART:**

**1. You are working in a logistics company responsible for delivering packages. Design a flowchart to manage the process of receiving, sorting, and delivering packages. Include decision structures for handling fragile items and urgent deliveries.**

Select

Read product name

Read Payment Method

Repeat

Select

If Payment Method is Cash

If Product name is valid

No No

Payment Method

Read AC. Number

Yes

Receive Payment

Yes

Print

Yes

Print Insufficient Funds

Insu t Funds

Until No More Product

No

If Sufficient Funds

Print Product

No

Yes

2. Imagine you are automating the process of a vending machine. Create a flowchart that

includes decision points for user input, selecting products, accepting payment, and dispensing

the correct item. Include error-handling for invalid inputs and insufficient funds.

No

Yes

No

Yes

No

Yes

Read Package

If Package is Fragile

Repeat

Receive package

If Package is valid

Package Status

Extra Safe packing

Delivery

Sort package

No

Yes

Print

Print Package

Urgent Delivery

If Delivery is Urgent

Until No More Package

**PSEUDOCODE:**

**1. Write pseudocode to find the smallest number among three given variables. Implement a**

**decision-making structure to compare the variables.**

1. Start.
2. Set a, b and c.
3. Input a, b and c.
4. IF (a<b and a<c)
5. Print “a is the smallest”
6. ELSE IF (b<a and b<c)
7. Print “b is the smallest”
8. ELSE
9. Print “c is the smallest”
10. End.

**2. Create pseudocode to subtract two numbers without using the - operator. (Hint: Use addition and complement techniques.)**

1. Start
2. Set a, b and j.
3. Set difference as 0.
4. Input a and b.
5. IF(a>b). //First we find the greater number
6. Set greatest=a.
7. Set smallest=b.
8. ELSE
9. Set greatest=b and smallest=a.
10. FOR j from smallest to greatest. //This loop will find the difference between both numbers.
11. Difference++
12. End FOR
13. IF(a<b). //This condition will help find the negative values also.
14. Difference=Difference\*(-1)
15. Print “Difference”.
16. End

**3. Develop pseudocode for a basic calculator that performs multiplication and division. The**

**pseudocode should prompt the user for two numbers and an operator, then display the**

**result of the operation.**

1. Start
2. Set a and b.
3. Input a and b.
4. Input operator.
5. IF (operator=” \*”)
6. Print a\*b.
7. ELSE If (operator=” /”)
8. Print a/b.
9. ELSE
10. Print “Invalid Operator”
11. End.

**ALGORITHM:**

**1. Write an algorithm to determine whether a number is a prime number. The algorithm should**

**iterate through possible divisors and determine if the number has any divisors other than 1 and itself.**

1. Start.
2. Input Number.
3. Initialize j as 2.
4. IF number is equal to 0 or 1.
5. Print “The Number is not prime”
6. IF (Number % j==0)
7. Print “Number is not prime” and Exit.
8. Increment j by 1 and repeat step 6 till j is less than the square root of number.
9. ELSE
10. Print “Number is prime”
11. End.

**2. Create an algorithm that asks the user for a day number (1-365) and outputs the**

**corresponding day of the week, assuming that January 1st is a Monday.**

1. Start.
2. Input the Number to determine the corresponding day.
3. IF (Number %2==0)
4. Print “The Day is Tuesday”
5. ELSE IF (Number %3==0)
6. Print “The Day is Wednesday”
7. ELSE IF (Number %4==0)
8. Print “The Day is Thursday”
9. ELSE IF (Number %5==0)
10. Print “The Day is Friday”
11. ELSE IF (Number %6==0)
12. Print “The Day is Saturday”
13. ELSE IF (Number %7==0)
14. Print “The Day is Sunday”
15. Else
16. Print “The Day is Monday”
17. End.

**3. Develop an algorithm for a program that takes two numbers as input and finds the Greatest**

**Common Divisor (GCD) of the two numbers using the Euclidean algorithm.**

1. Start.
2. Input a and b.
3. While b! = 0
4. Store the value of b in random.
5. Set b to a % b.
6. Update a to random.
7. Print “GCD= a”
8. End.