# Flowchart & Algorithm of 1-35 programs

Name:-

Sem:-

Roll no:-

Reg nO:-

Subject:-

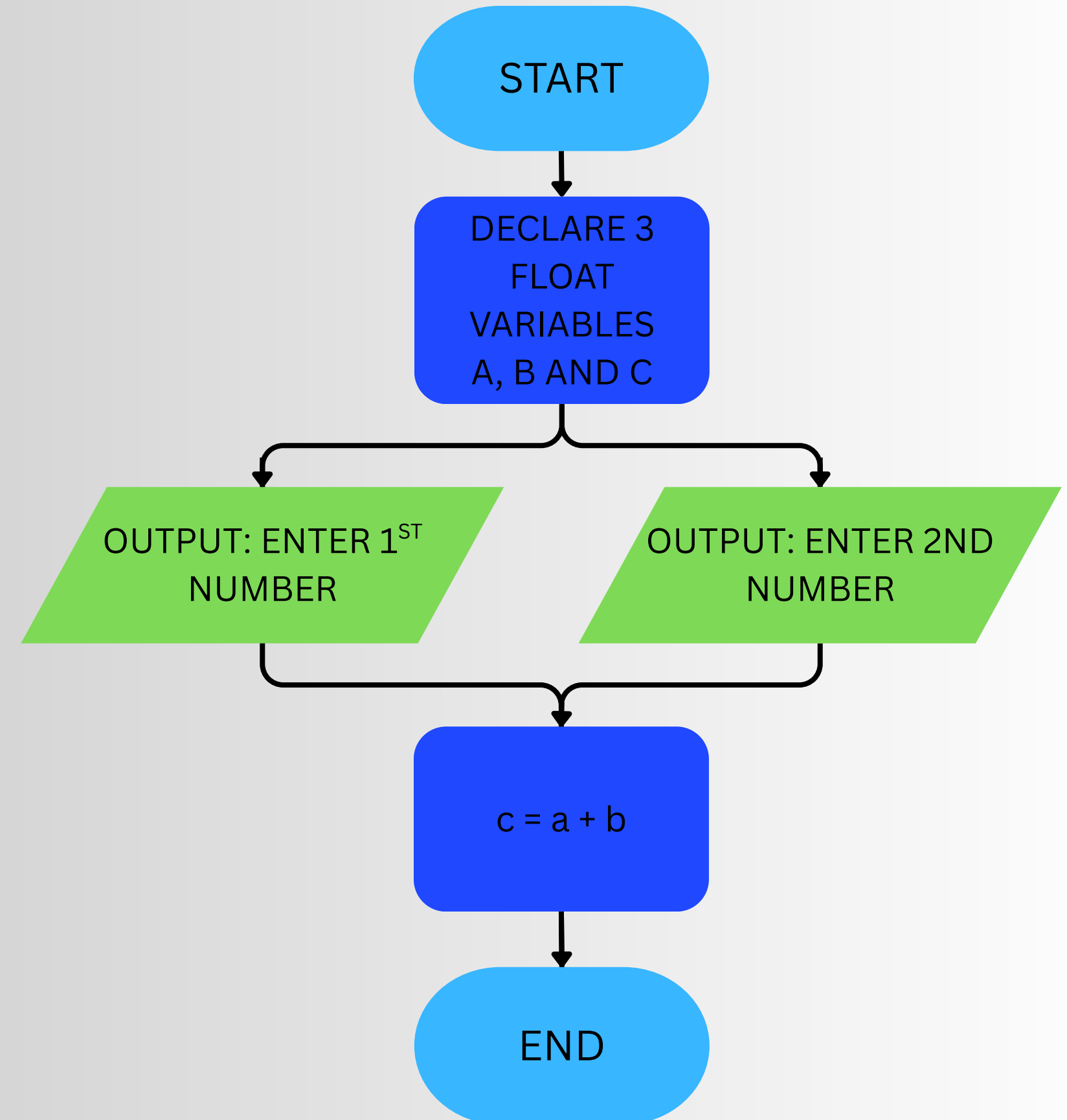## Q1. Print Adamas University

Alogorithm:-

1. Start
2. Include the header file <stdio.h> to allow use of input/output functions.
3. Begin the main() function (program execution starts here).
4. Use the printf() function to display the text "Adamas University" on the screen.
5. Return 0 to indicate successful program execution.
6. End

Start

Output: Adamas University
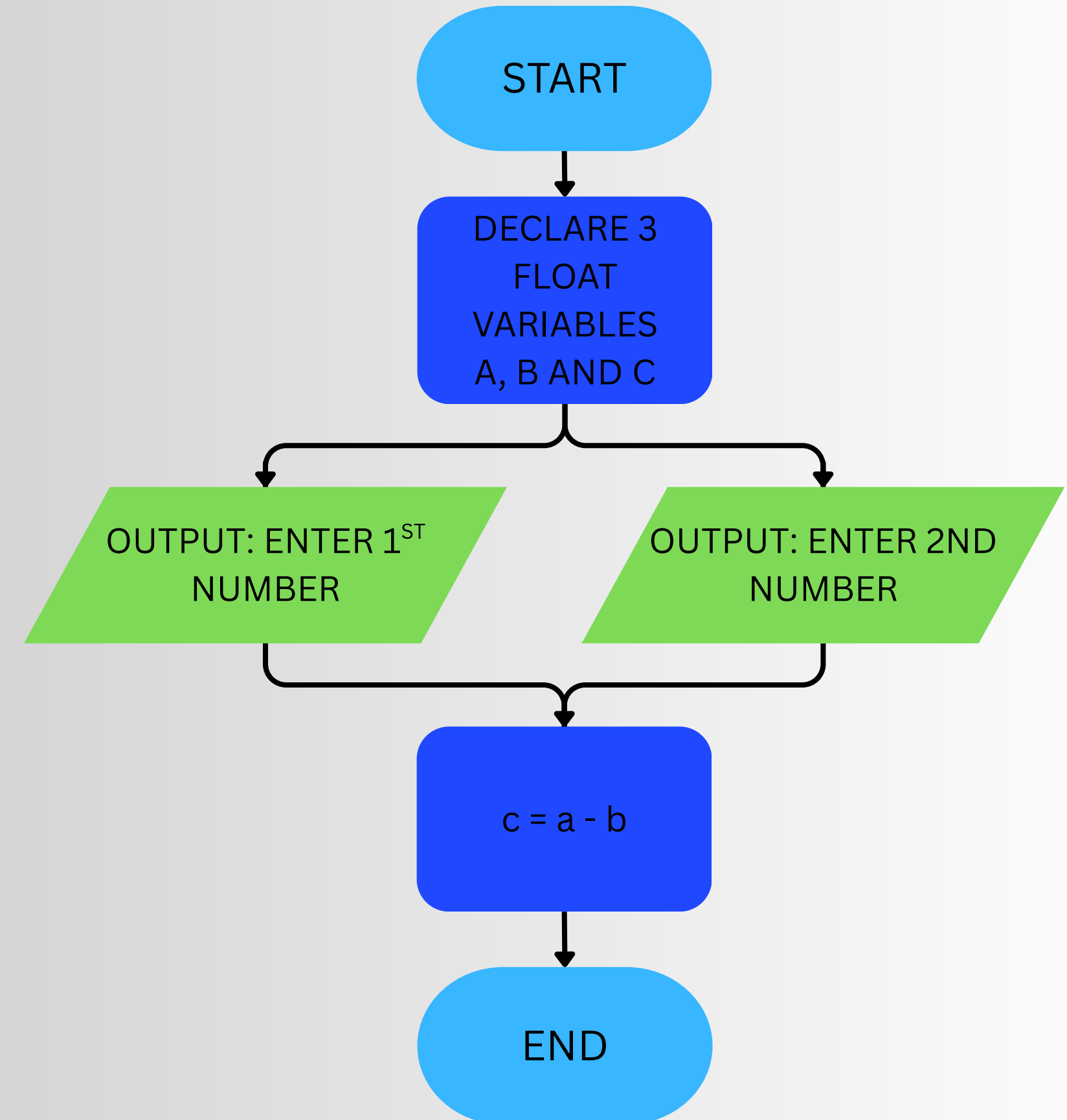
End

2
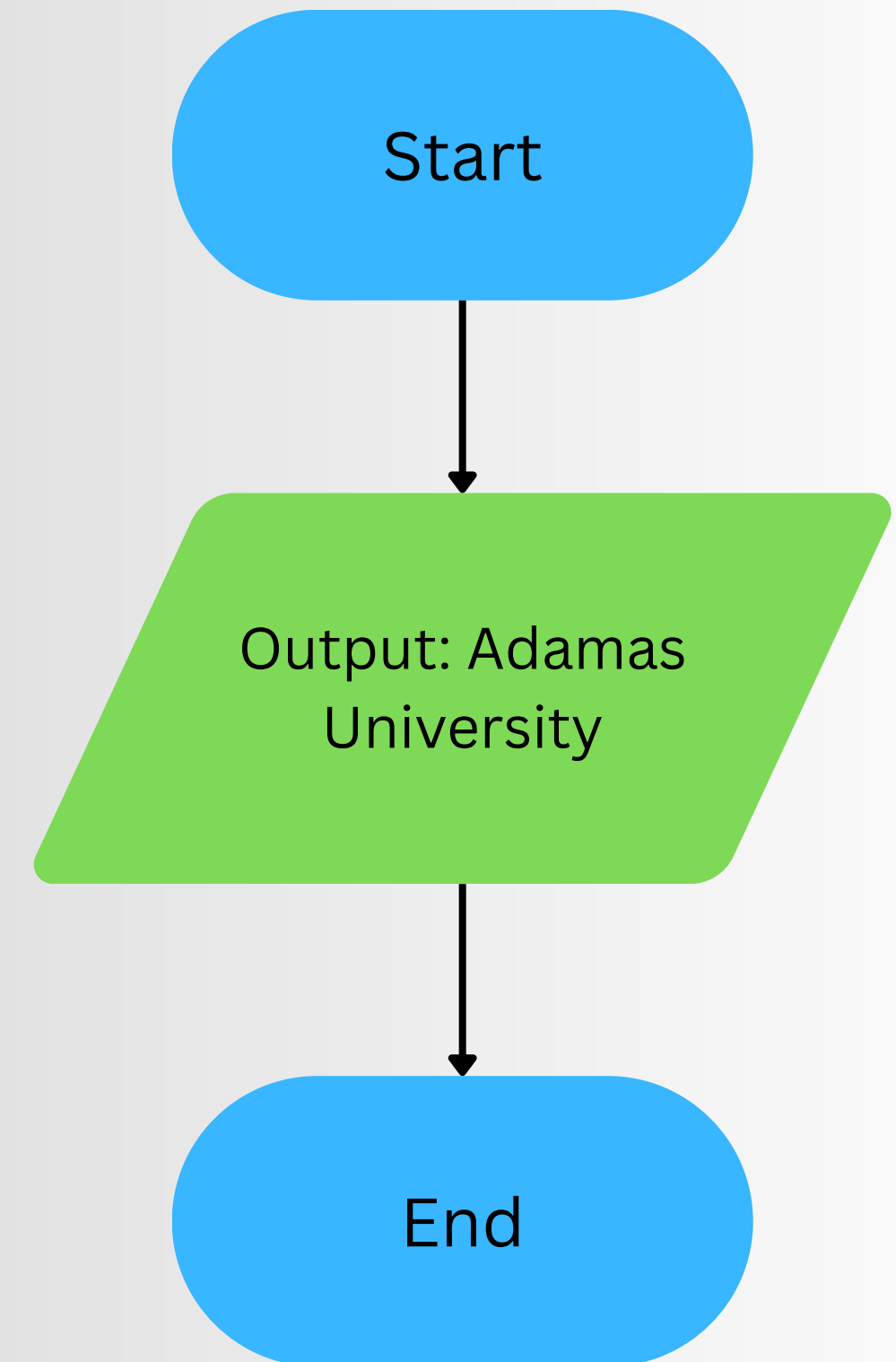
## Q2. Addition of 2 numbers

Algorithm:-

1. Start
2. Declare three variables a, b, and c of type float.
3. Display the message: "Enter two numbers:".
4. Read the values of a and b from the user.
5. Calculate the sum: c = a + b.
6. Display the result: "sum is : " followed by the value of c.
7. End

START

DECLARE 3 FLOAT VARIABLES A, B AND C

OUTPUT: ENTER 1ST NUMBER

OUTPUT: ENTER 2ND NUMBER

c = a + b

END

## Q3. **Difference of 2 numbers**

Algorithm:-

1. Start
2. Declare three variables a, b, and c of type float.
3. Display the message: "Enter two numbers:".
4. Read the values of a and b from the user.
5. Calculate the difference: c = a - b.
6. Display the result: "Difference is: " followed by the value of c.
7. End

START → DECLARE 3 FLOAT VARIABLES A, B AND C → OUTPUT: ENTER 1ST NUMBER / OUTPUT: ENTER 2ND NUMBER → c = a - b → END
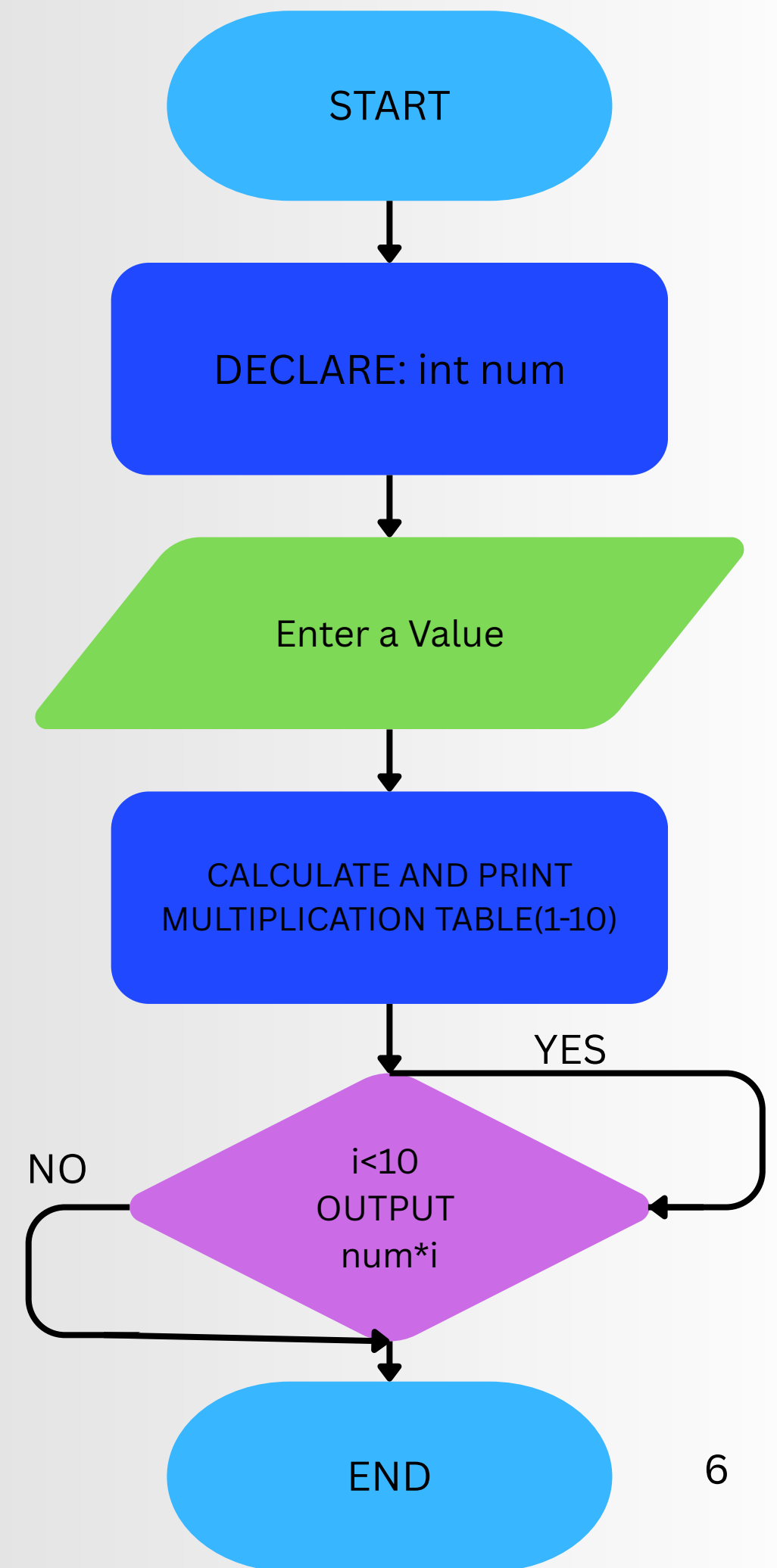
## Q4. ASCII of a character

Algorithm:-

1. Start
2. Declare a variable c of type char.
3. Display the message: "Enter a character:".
4. Read a character from the user and store it in c.
5. Display the message: "The ASCII value is: " followed by the integer value of c.
6. End

START

DECLARE: char ch

OUTPUT: Enter a Character

OUTPUT

END
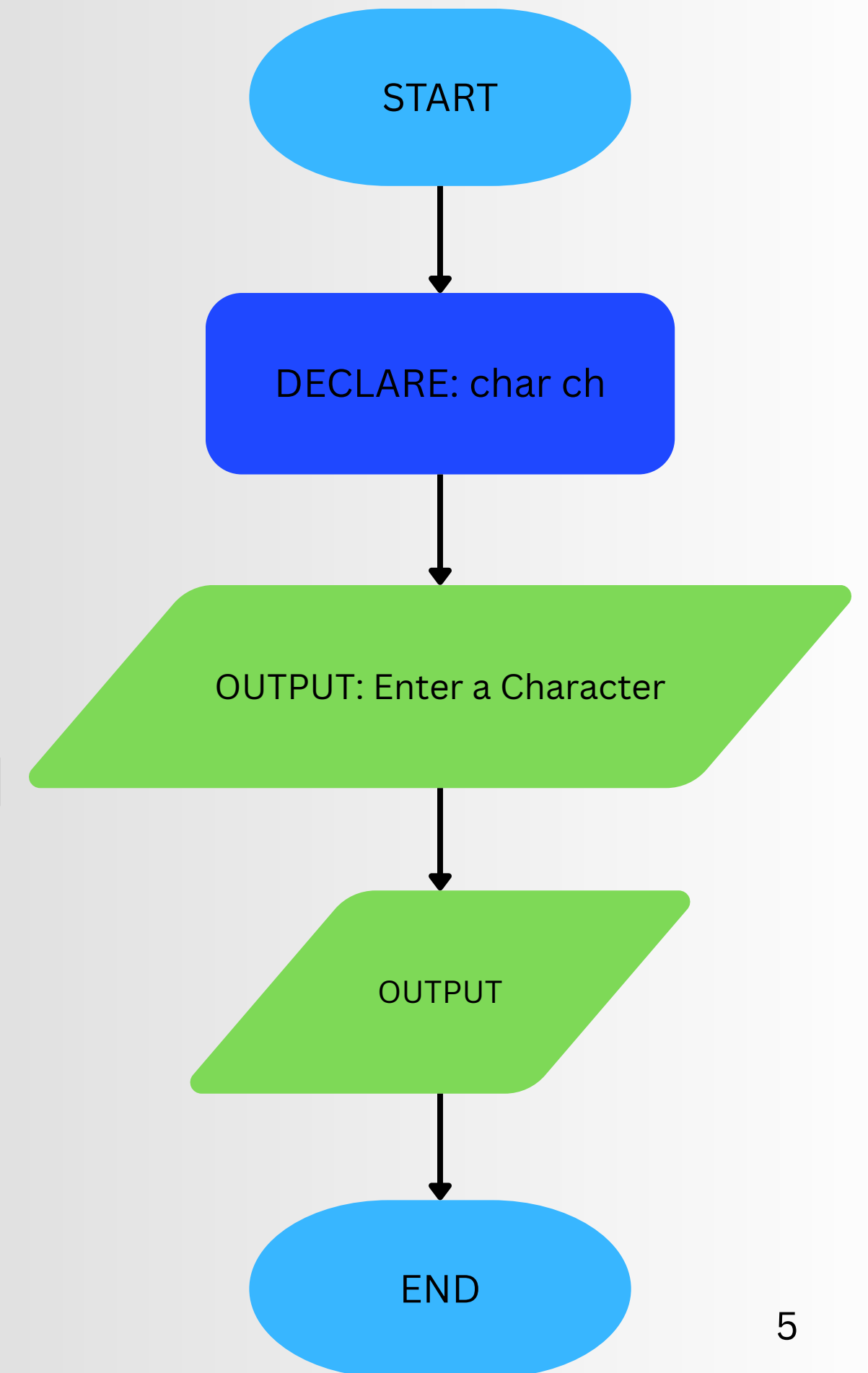
## Q5. Print Multiplication Table withour using For loop

Algorithm:-

1. Start
2. Declare num = 5.
3. Print "Multiplication table of 5:".
4. Print each product from num*1 to num*10 manually.
5. End

```
START
  ↓
DECLARE: int num
  ↓
Enter a Value
  ↓
CALCULATE AND PRINT
MULTIPLICATION TABLE(1-10)
  ↓
        YES
i<10
OUTPUT
num*i
NO
  ↓
END
```

6

## Q6. Calculate Simple Interest

Algorithm:-

1. Start
2. Declare p, r, t, and si as float.
3. Prompt the user: "Enter principal, rate, and time:".
4. Read values of p, r, and t.
5. Calculate simple interest: si = (p * r * t) / 100.
6. Display "Simple Interest = " followed by si.
7. End

START

DECLARE p, r, t and si as float

si = (p * r * t)/100

Display si

END
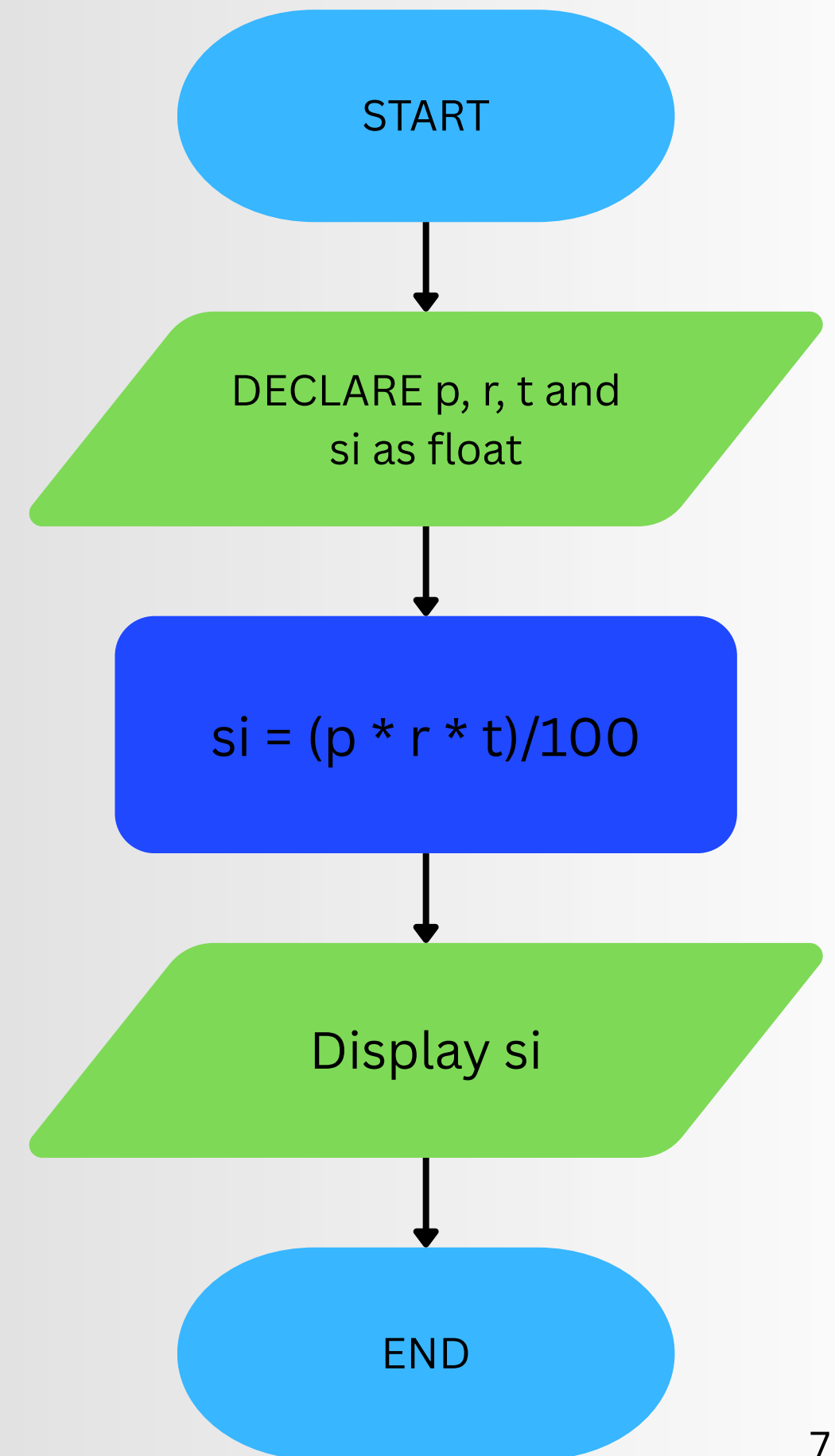
## Q7. Calculate the area of a circle

Algorithm:-

1. Start
2. Declare radius and area as float.
3. Prompt the user: "Enter the radius of the circle:".
4. Read the value of radius.
5. Calculate area: area = 3.14 * radius * radius.
6. Display "Area of the circle = " followed by area.
7. End

START

DECLARE: Input radius as r and area
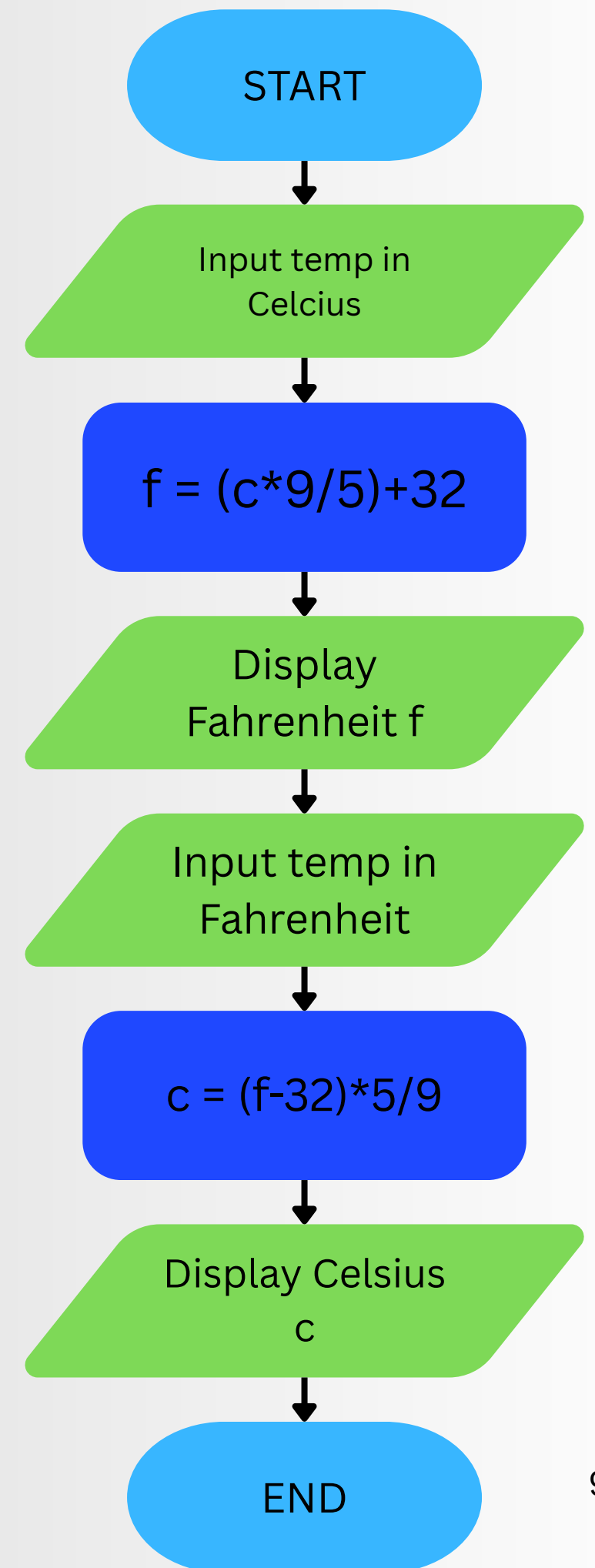
area = 3.14 * r * r

Display area

END
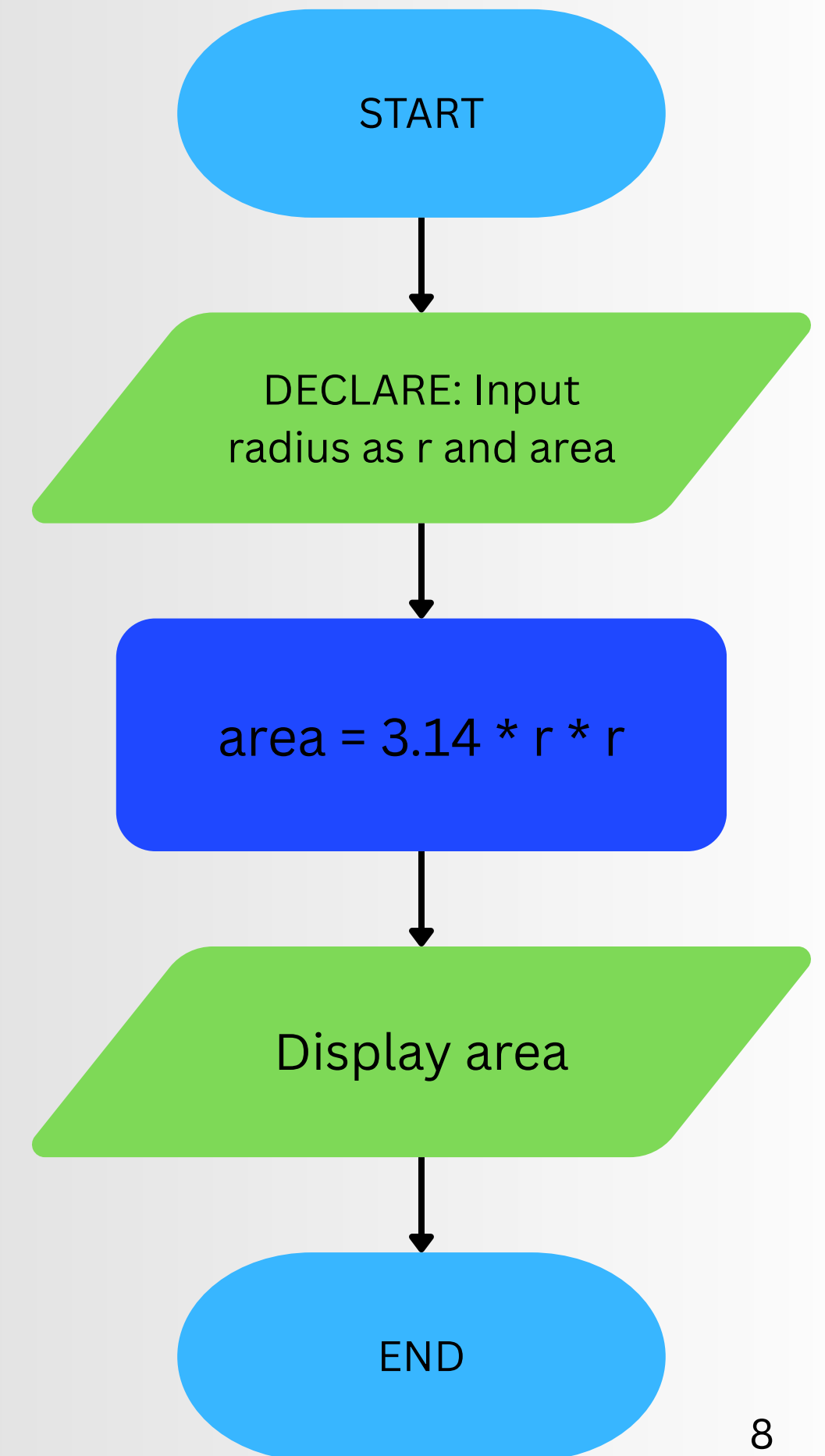
**Q8. Convert temperatures between Celsius and Fahrenheit and display both conversions.**

Algorithm:-

1. Start
2. Declare c, f, temp as float.
3. Prompt the user: "Enter temperature in Celsius:".
4. Read value of c.
5. Convert to Fahrenheit: f = (c * 9 / 5) + 32.
6. Display "Temperature in Fahrenheit = " followed by f.
7. Prompt the user: "Enter temperature in Fahrenheit:".
8. Read value of f.
9. Convert to Celsius: c = (f - 32) * 5 / 9.
10. Display "Temperature in Celsius = " followed by c.
11. End

START

Input temp in Celcius

f = (c*9/5)+32

Display Fahrenheit f

Input temp in Fahrenheit

c = (f-32)*5/9

Display Celsius c

END

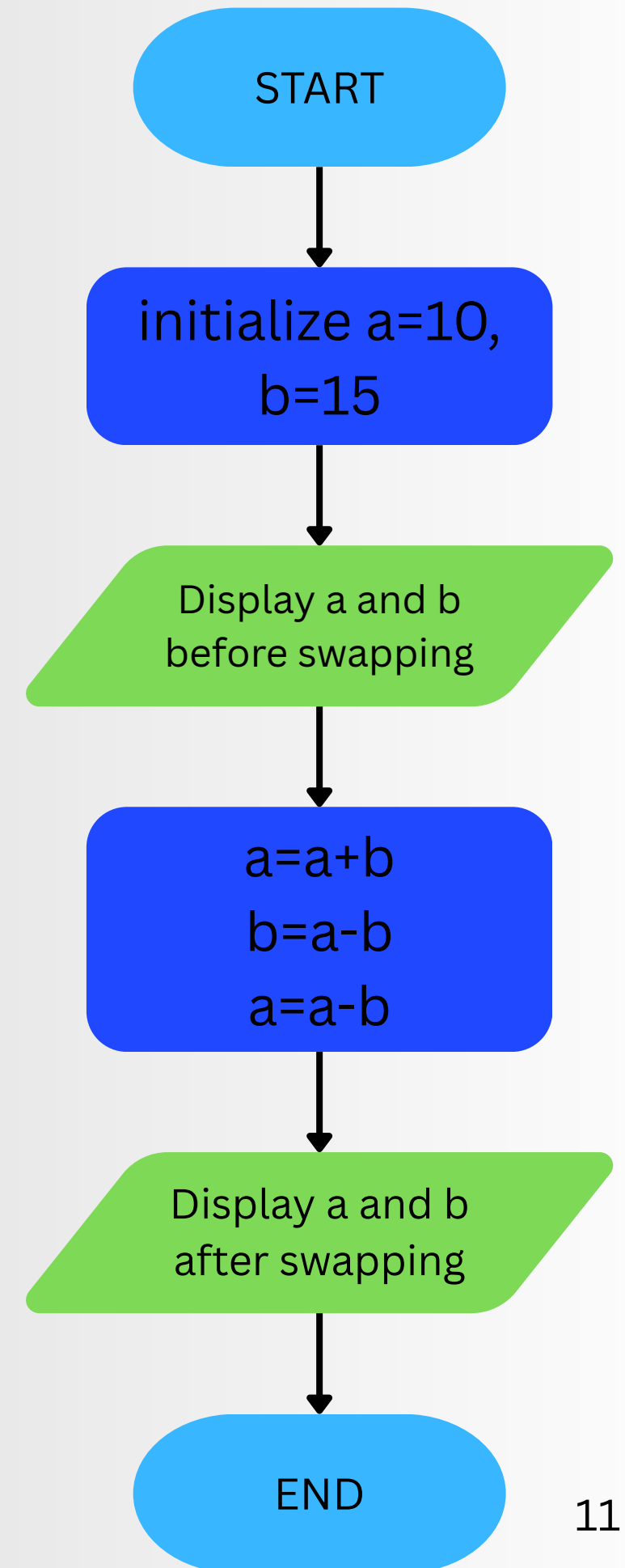## Q9. Swapping 2 numbers using third variable

Algorithm:-

1. Start
2. Declare integers a, b, and c.
3. Initialize a = 10 and b = 15.
4. Display "Before swapping: a = 10, b = 15".
5. Store a in c.
6. Assign b to a.
7. Assign c to b.
8. Display "After swapping: a = 15, b = 10".
9. End

START

initialize a=10, b=15

Display a and b before swapping

c=a
a=b
b=c

Display a and b after swapping

END

10
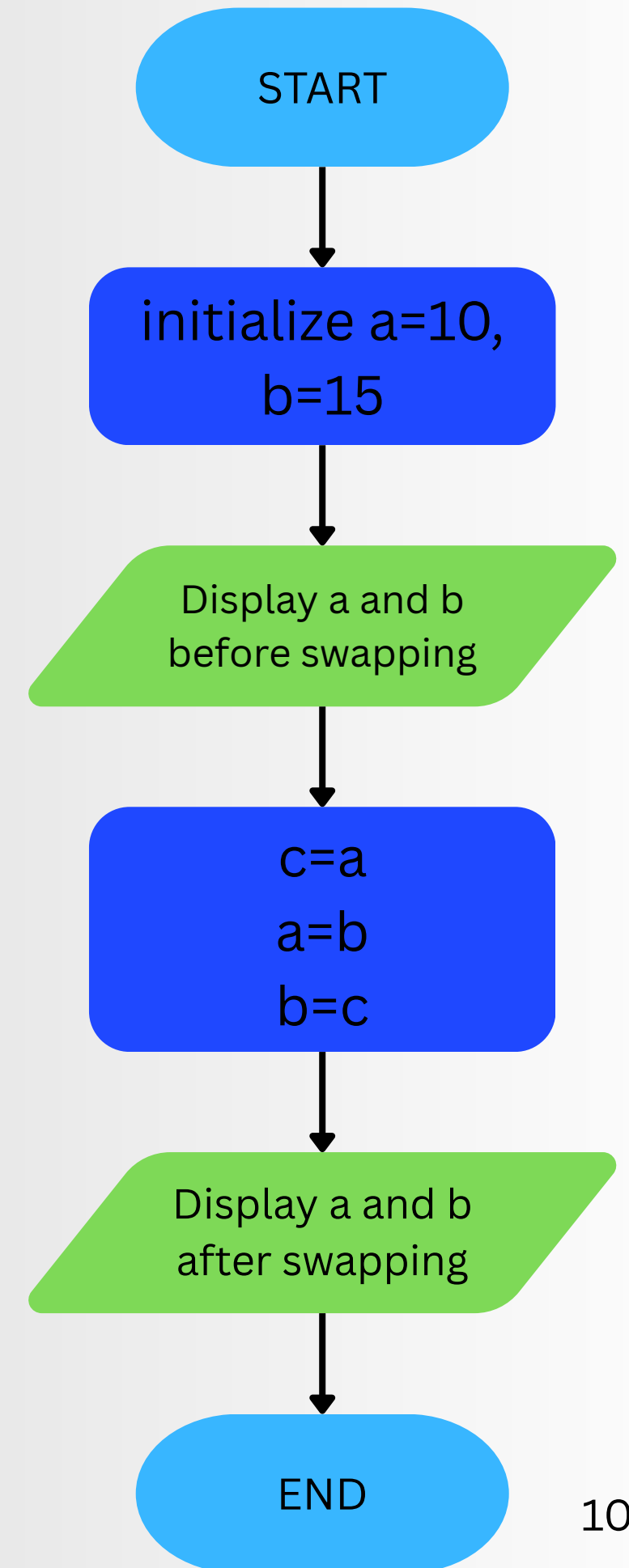
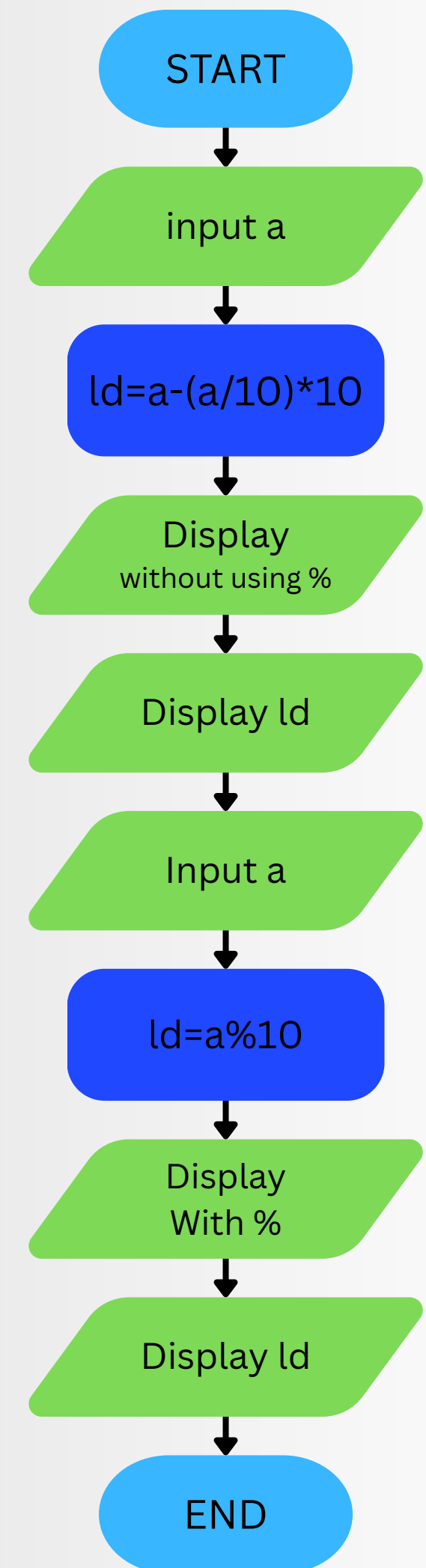## Q10. Swapping 2 numbers without using third variable

Algorithm:-

1. Start
2. Declare integers a and b.
3. Initialize a = 10 and b = 15.
4. Display "Before swapping: a = 10, b = 15".
5. Add a and b, store in a → a = a + b.
6. Subtract new b from a, store in b → b = a - b.
7. Subtract new b from a, store in a → a = a - b.
8. Display "After swapping: a = 15, b = 10".
9. End

START

initialize a=10, b=15

Display a and b before swapping

a=a+b
b=a-b
a=a-b

Display a and b after swapping

END

11

## **Q11. Display Last Digit with and without using the %**

Algorithm:-

1. Start
2. Declare integers a and ld.
3. Prompt: "Enter the value of a:"
4. Read the value of a.
5. Find last digit without modulus: ld = a - (a / 10) * 10.
6. Display "Last digit (without modulus):" followed by ld.
7. Prompt: "Enter the value of a:" again.
8. Read the value of a.
9. Find last digit using modulus: ld = a % 10.
10. Display "Last digit (using modulus):" followed by ld.
11. End



START

input a

ld=a-(a/10)*10

Display
without using %

Display ld

Input a

ld=a%10

Display
With %

Display ld

END

12

## Q12. Calculate Compound Interest

Algorithm:-

1. Start
2. Declare variables p, r, t, amt, and ci as double.
3. Prompt: "Enter Principle Amount:", read p.
4. Prompt: "Enter Rate:", read r.
5. Prompt: "Enter Time:", read t.
6. Calculate total amount: amt = p * pow((1 + r / 100), t).
7. Calculate compound interest: ci = amt - p.
8. Display "Compound Interest =" followed by ci.
9. Display "Total Amount =" followed by amt.
10. End



13
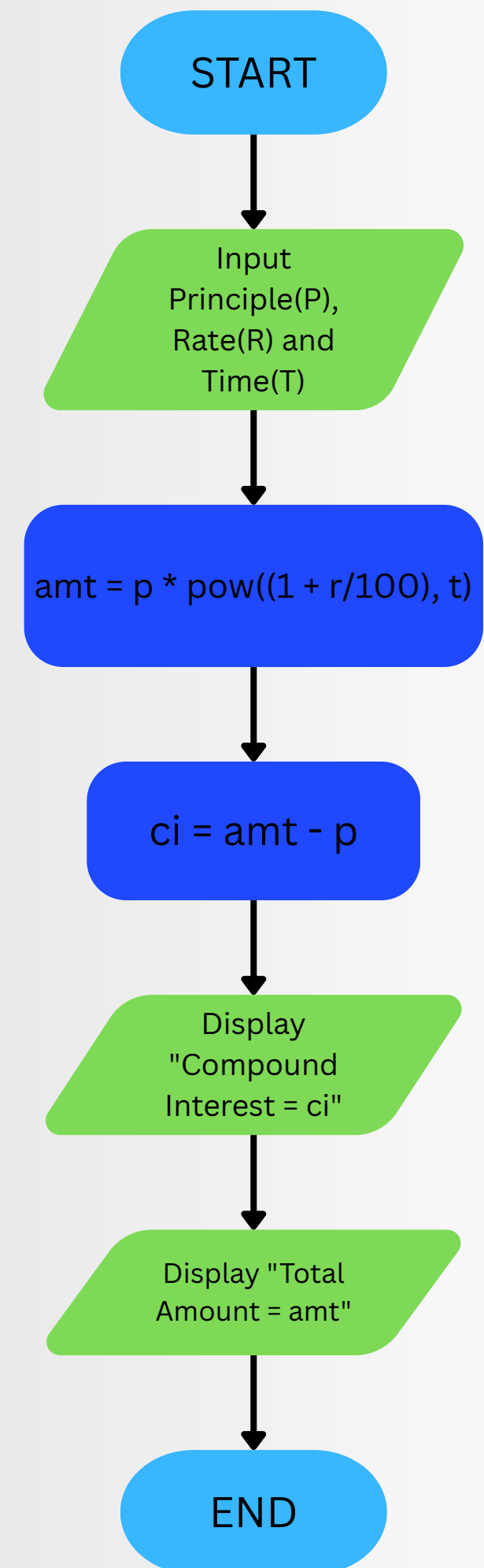
**Q13. Calculate Area and Perimeter of a Rectangle**
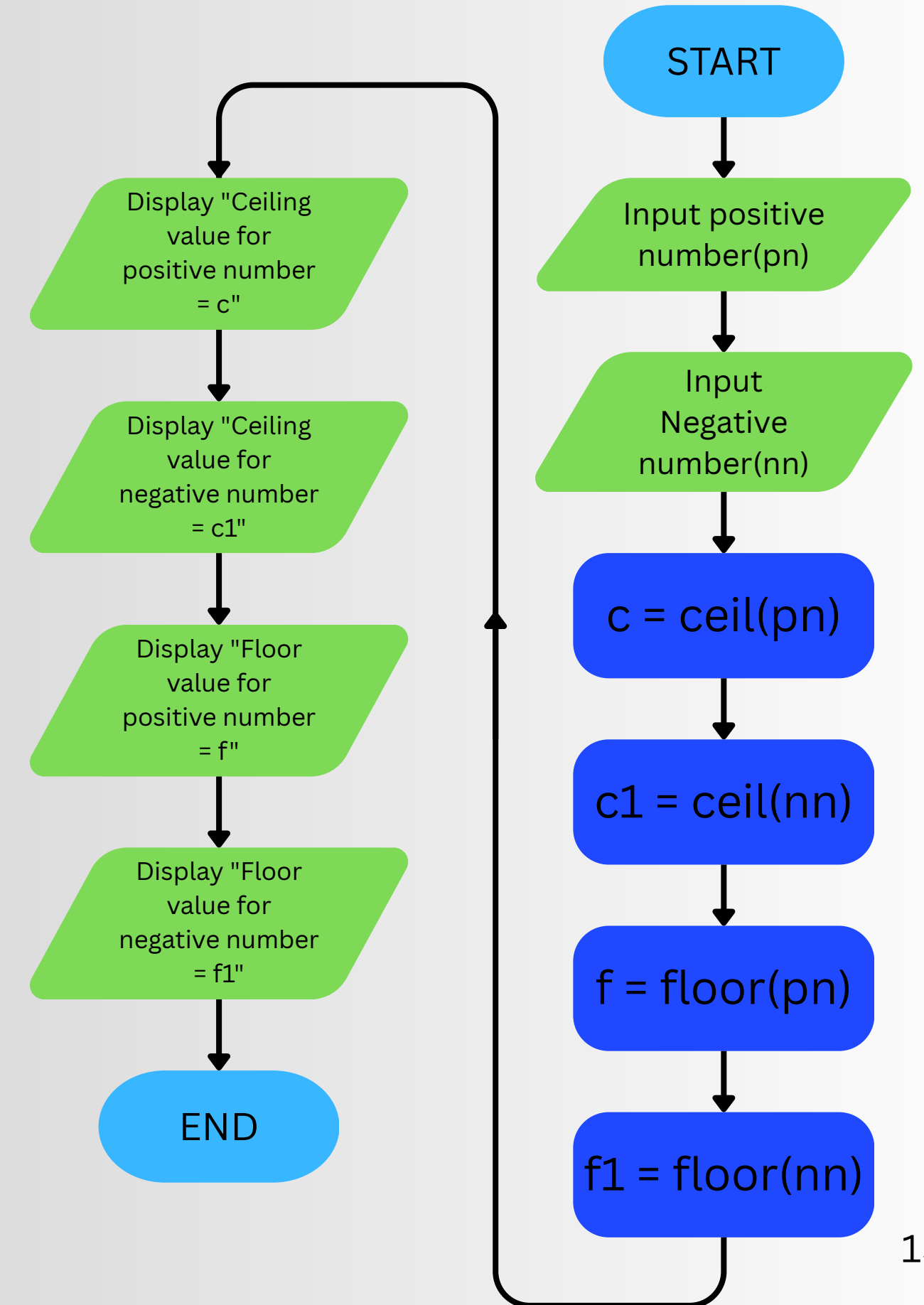
Algorithm:-

1. Start
2. Declare variables l, b, a, and p as float.
3. Prompt: "Enter the length of the Rectangle:", read l.
4. Prompt: "Enter the breadth of the Rectangle:", read b.
5. Calculate area: a = l * b.
6. Calculate perimeter: p = 2 * (l + b).
7. Display "Area =" followed by a.
8. Display "Perimeter =" followed by p.
9. End



START → Input Length(l) and Breadth(b) → a=l*b → p=2(l+b) → Display "Area = a" → Display "Perimeter = p" → END

14
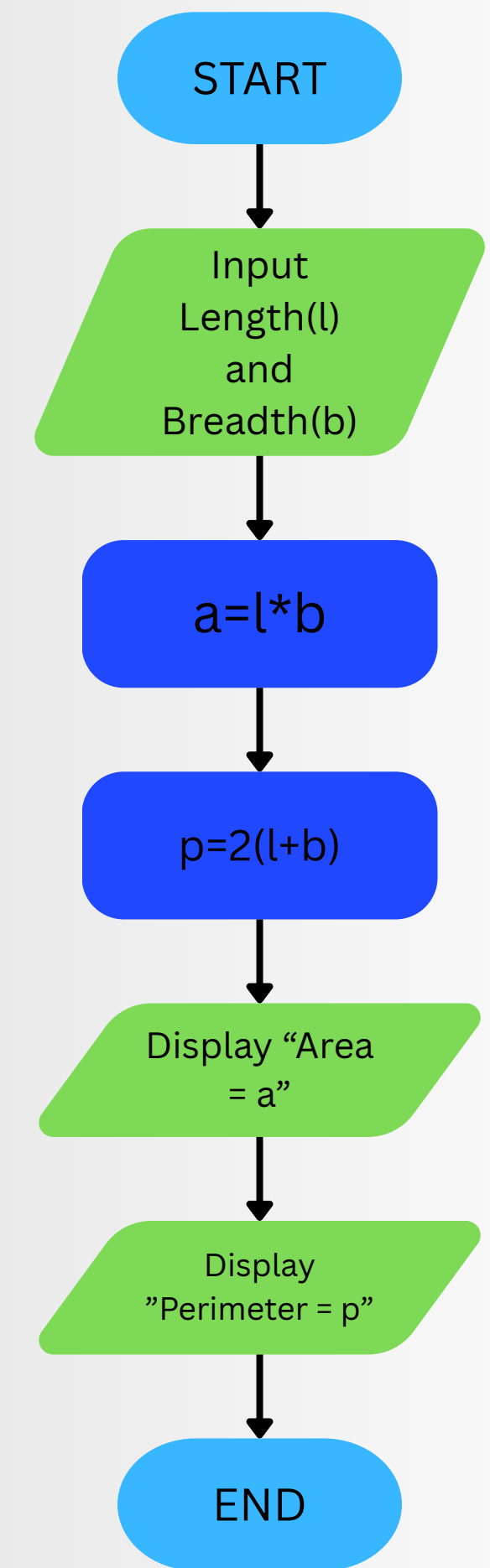
# Q14. Print the Floor and Ceil of a number
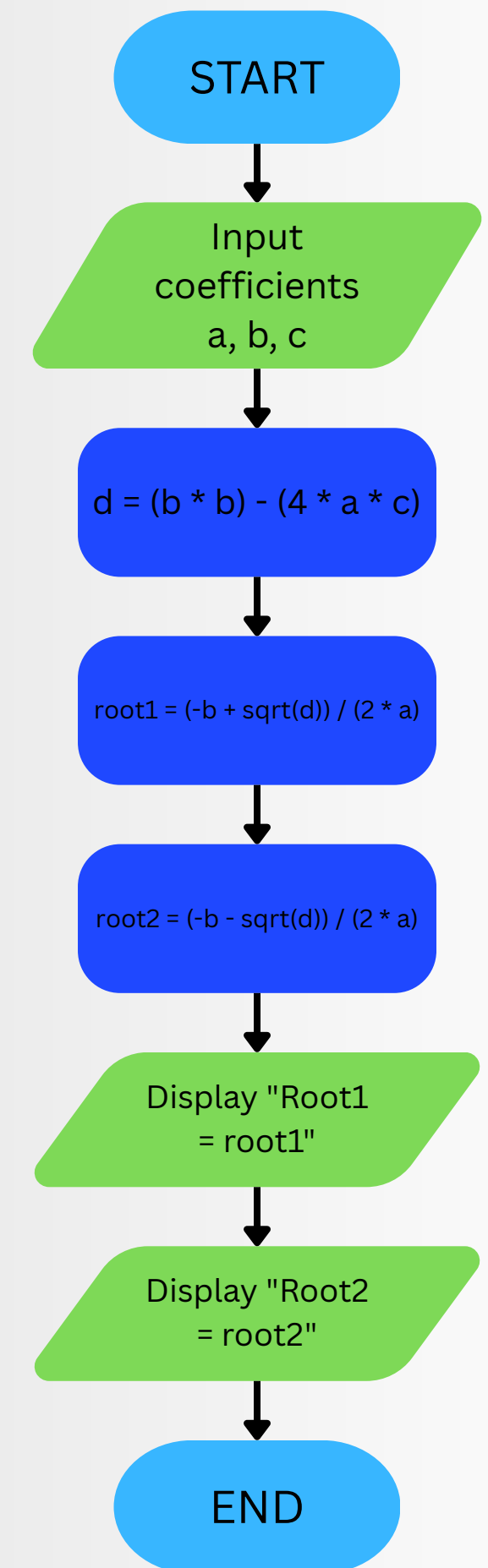
Algorithm:-

1. Start
2. Declare variables pn, nn, c, c1, f, f1 as float.
3. Prompt: "Enter a positive number:", read pn.
4. Prompt: "Enter a negative number:", read nn.
5. Calculate ceiling of positive: c = ceil(pn).
6. Calculate ceiling of negative: c1 = ceil(nn).
7. Calculate floor of positive: f = floor(pn).
8. Calculate floor of negative: f1 = floor(nn).
9. Display all four values (c, c1, f, f1).
10. End



15

**Q15. Find the Roots of an Quadratic equation**

Algorithm:-

1. Start
2. Declare variables a, b, c, d, root1, and root2 as float.
3. Prompt: "Enter coefficients a, b and c:", read a, b, and c.
4. Calculate discriminant: d = b * b - 4 * a * c.
5. Calculate first root: root1 = (-b + sqrt(d)) / (2 * a).
6. Calculate second root: root2 = (-b - sqrt(d)) / (2 * a).
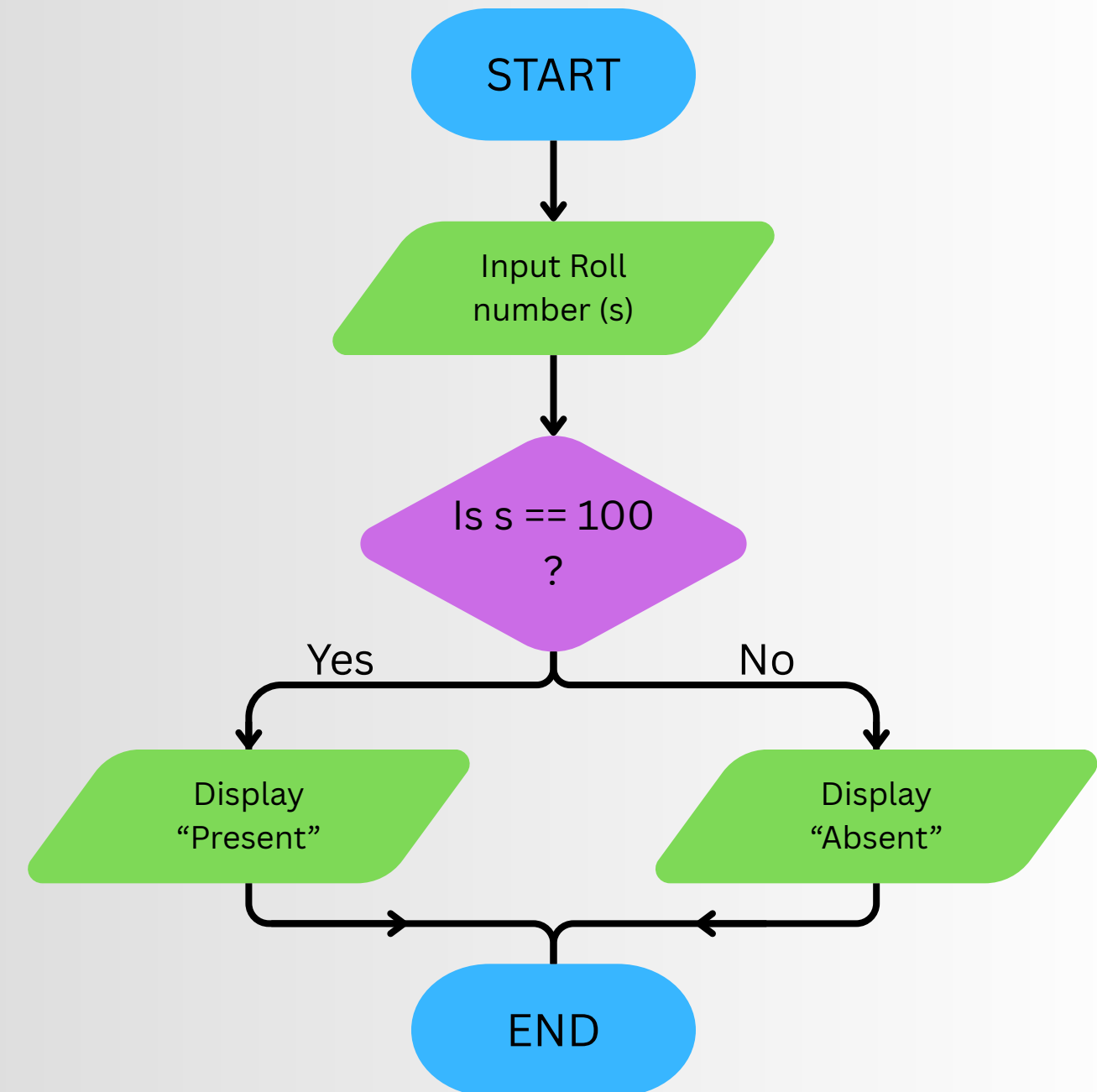7. Display root1 and root2.
8. End



START

Input coefficients a, b, c

d = (b * b) - (4 * a * c)

root1 = (-b + sqrt(d)) / (2 * a)

root2 = (-b - sqrt(d)) / (2 * a)

Display "Root1 = root1"

Display "Root2 = root2"

END

16

## Q16. Check whether the student with a 100 roll no is absent or present

Algorithm:-
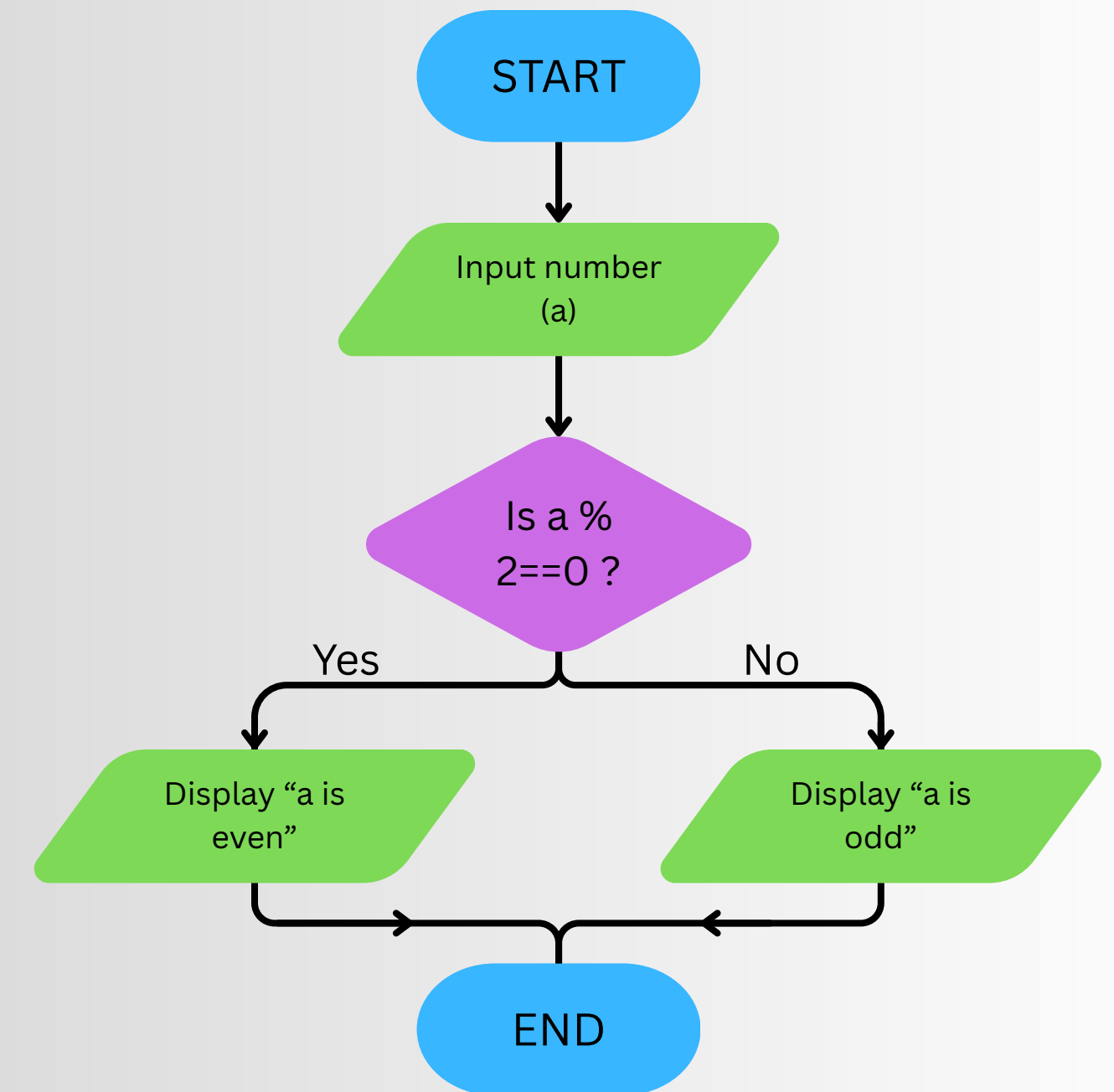
1. Start
2. Declare an integer variable s.
3. Prompt: "Enter the Roll number:", read s.
4. If s == 100, display "Present".
5. Else, display "Absent".
6. End

START

Input Roll number (s)

Is s == 100 ?

Yes

No

Display "Present"

Display "Absent"

END

ADAMAS
UNIVERSITY
KOLKATA
PURSUE EXCELLENCE

17

**Q17. Check whether a number is odd or even**

Algorithm:-

1. Start
2. Declare an integer variable a.
3. Prompt: "Enter any number:", read a.
4. If a % 2 == 0, display a is Even.
5. Else, display a is Odd.
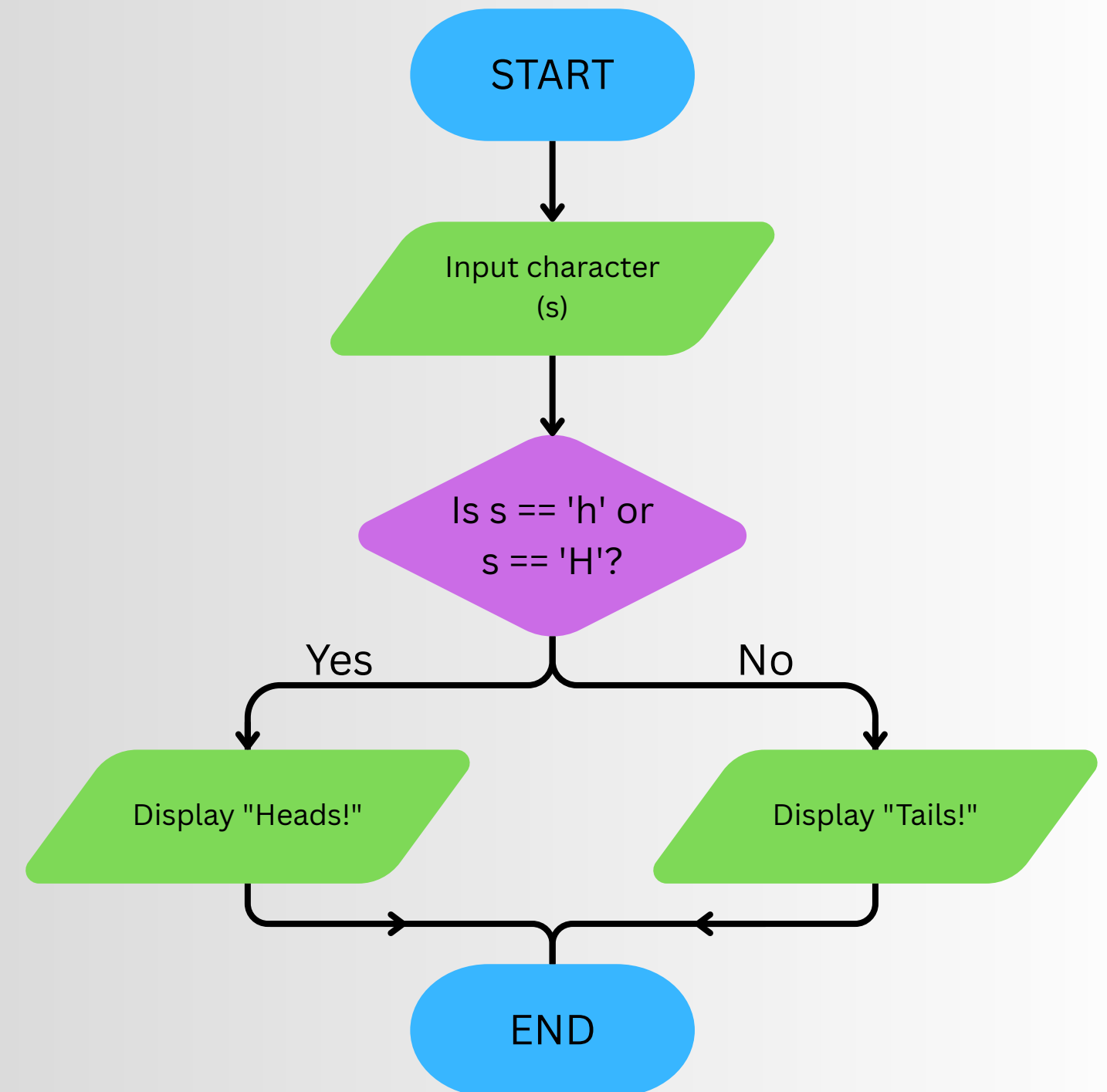6. End

**Q18. Find the greater among 2 numbers**

Algorithm:-

1. Start
2. Declare two integer variables a and b.
3. Prompt: "Enter two numbers:", read a and b.
4. If a > b, display a is Greater.
5. Else, display b is Greater.
6. End
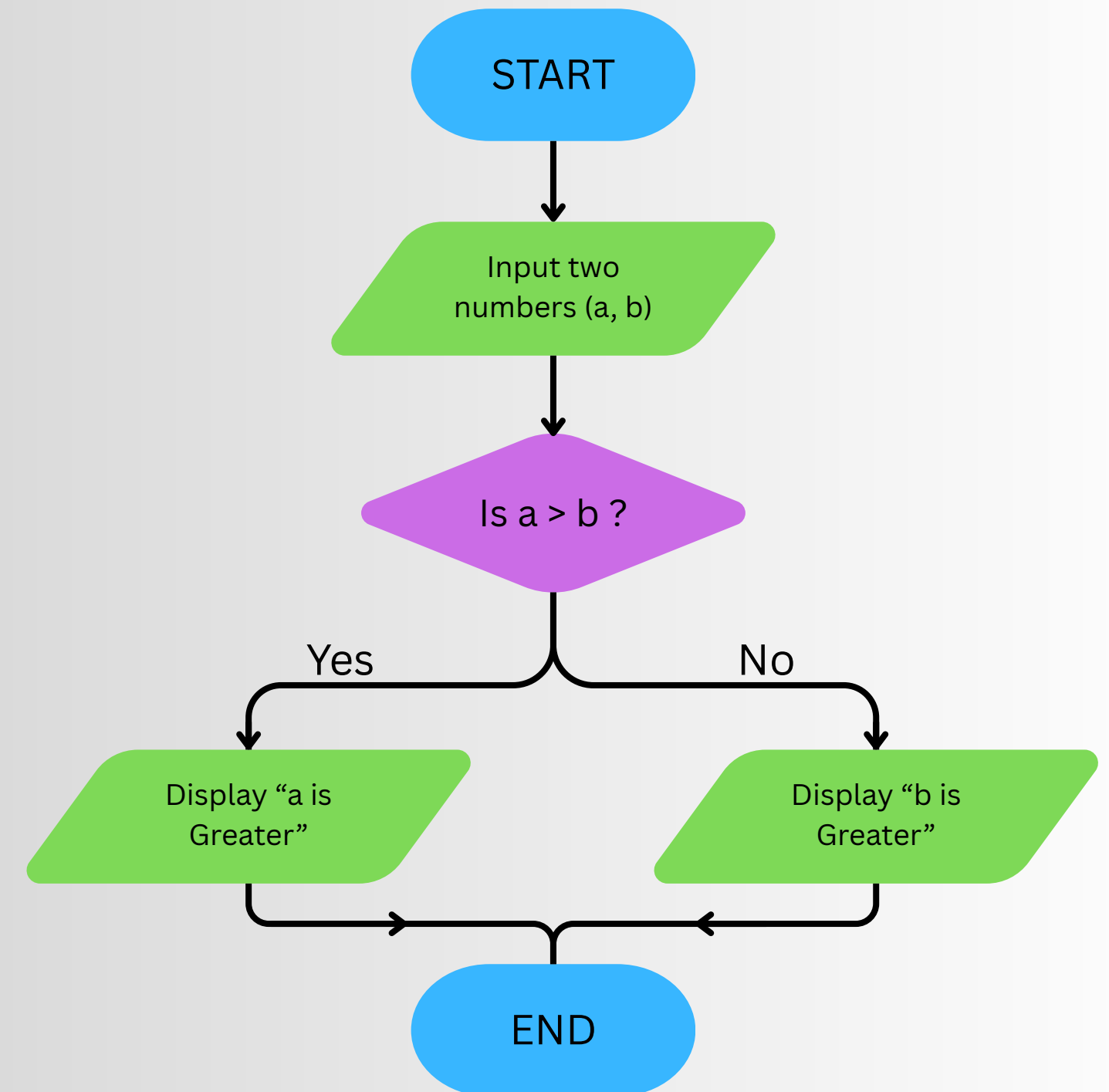
## Q19. Check for Heads or Tails

Algorithm:-

1. Start
2. Declare a character variable s.
3. Prompt: "Enter Head or Tails:", read s.
4. If s == 'h' or s == 'H', display "Heads!".
5. Else, display "Tails!".
6. End

**Q20. Check whether a number is a whole number**

Algorithm:-

1. Start
2. Declare an integer variable n.
3. Prompt: "Enter a number:", read n.
4. If n > 0, display n is a whole number.
5. Else, display n is not a whole number.
6. End

START

Input number (n)

Is n <= 0 ?

Yes     No

Display "n is not a whole number"

Display "n is a whole number"

END
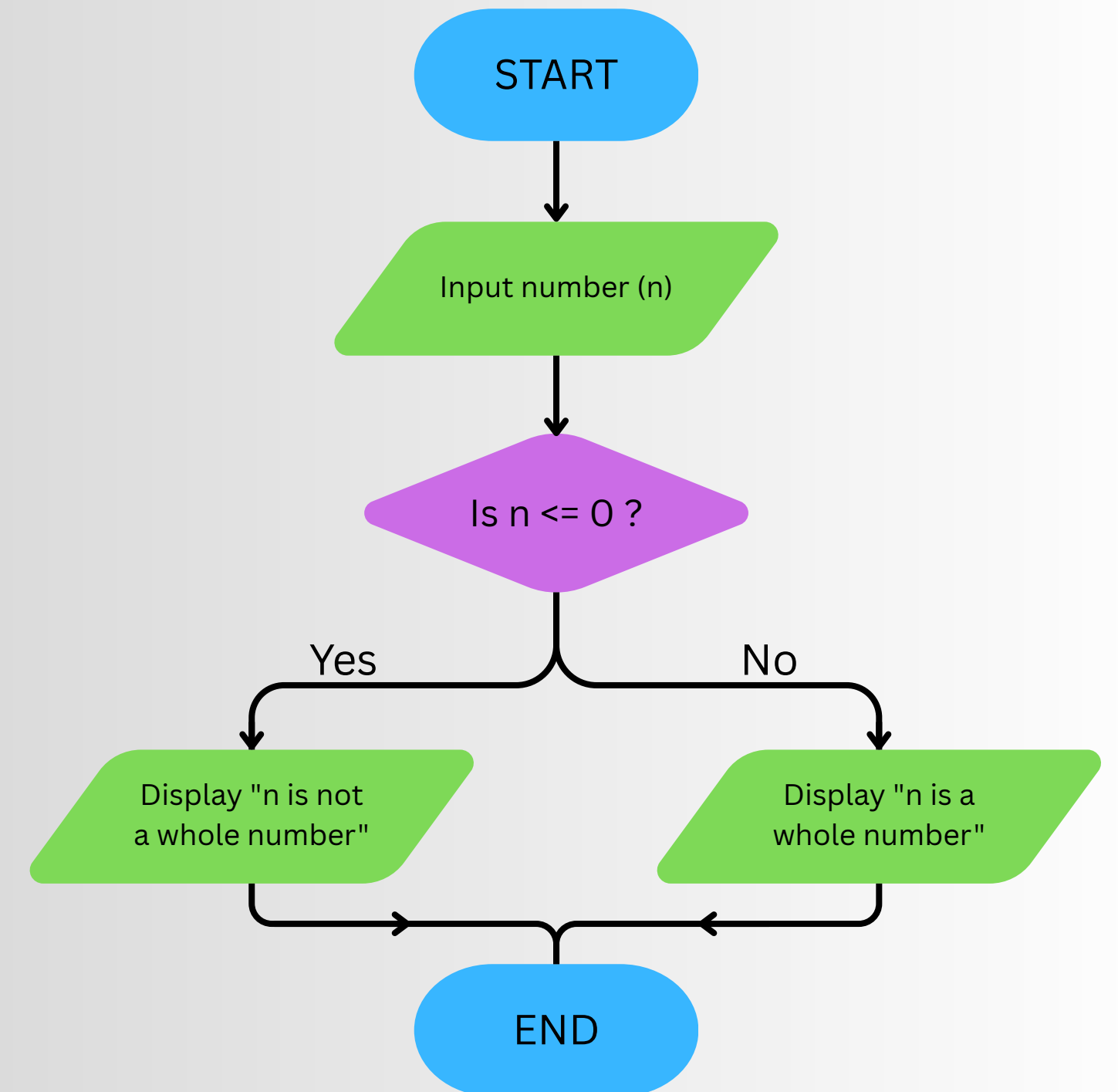
## Q21. Check whether a year is a leap year

Algorithm:-
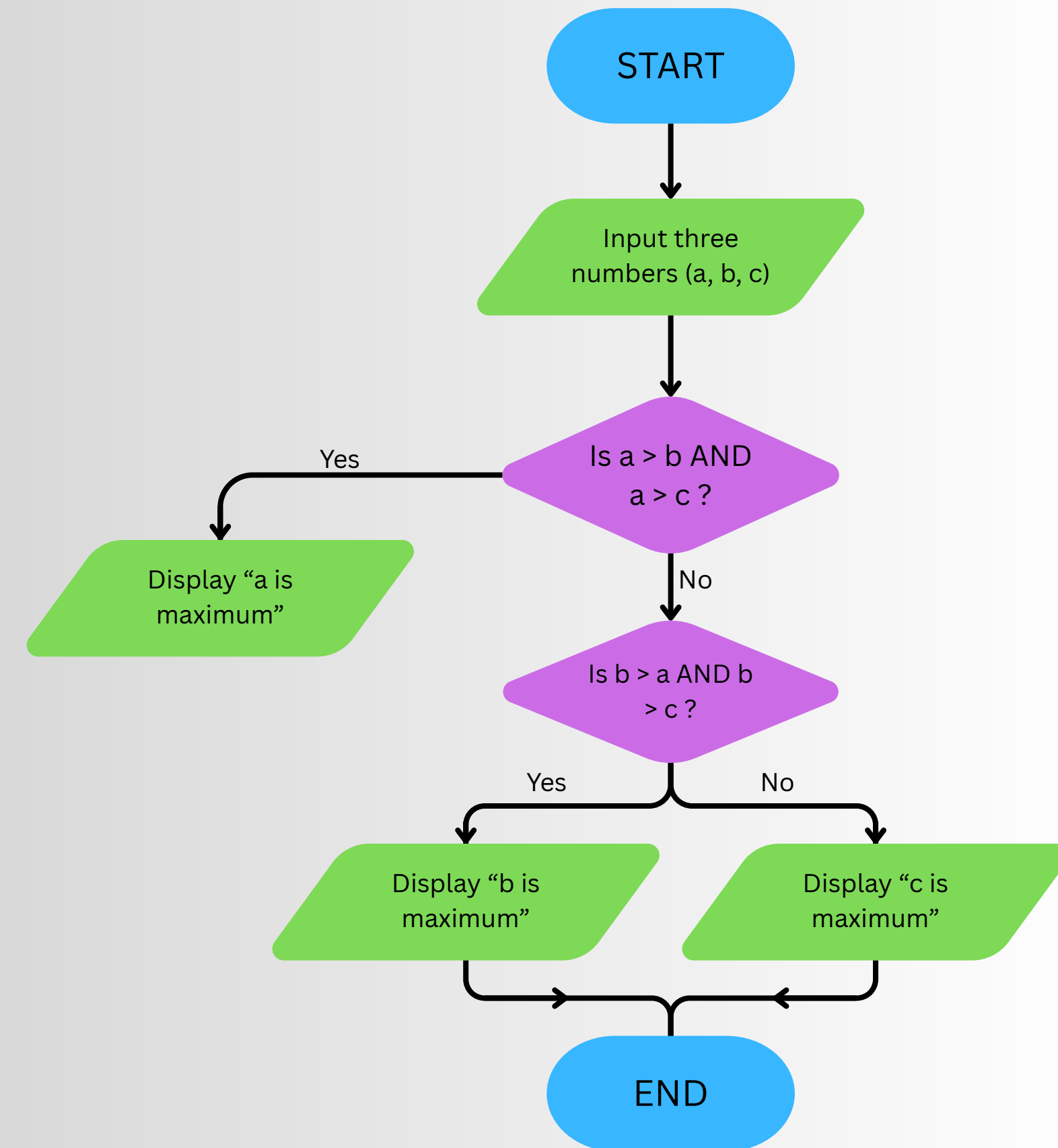
1. Start
2. Declare an integer variable year.
3. Prompt: "Enter a year:", read year.
4. If year % 400 == 0, display "year is a leap year".
5. Else if year % 100 == 0, display "year is not a leap year".
6. Else if year % 4 == 0, display "year is a leap year".
7. Else, display "year is not a leap year".
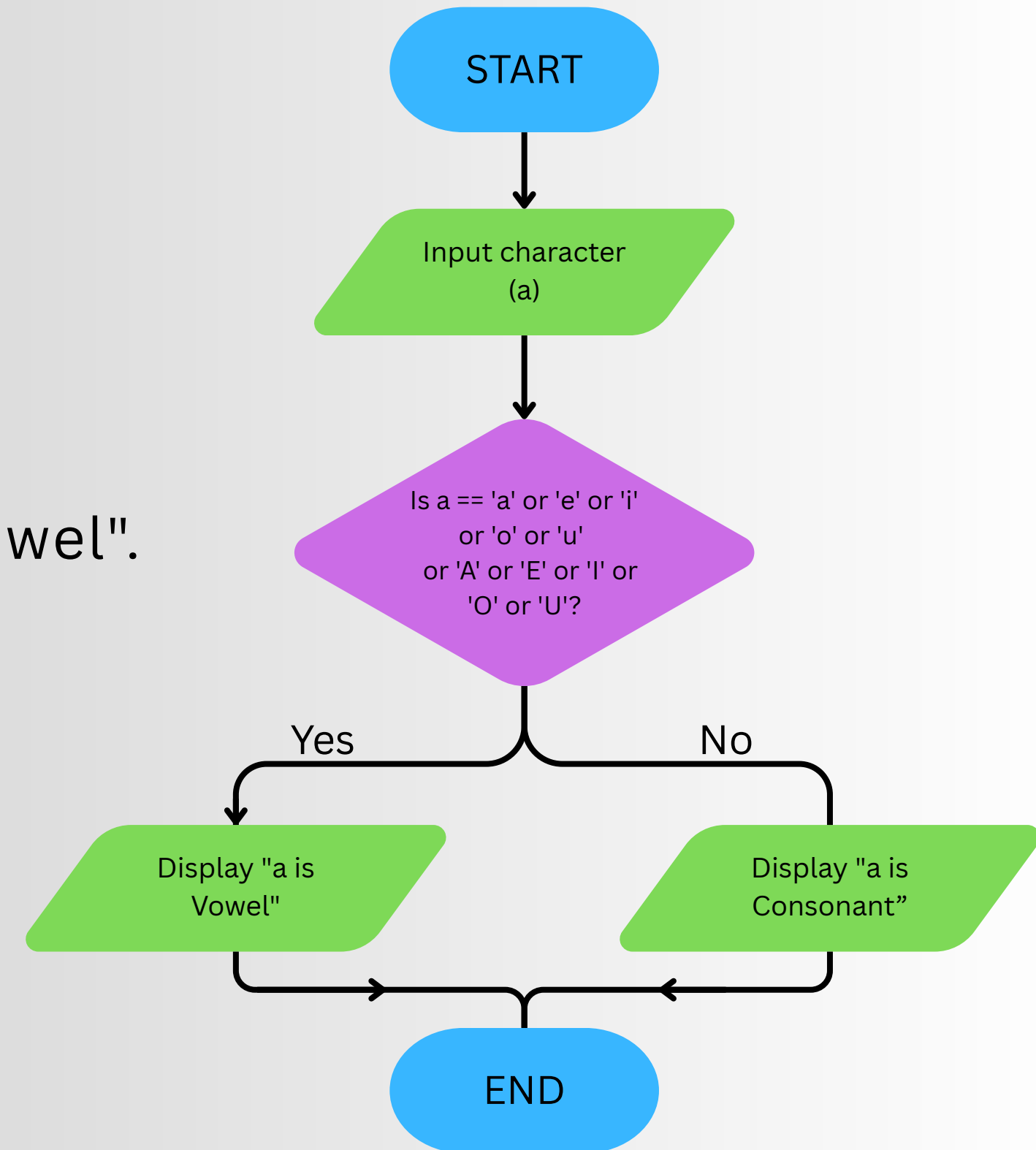8. End
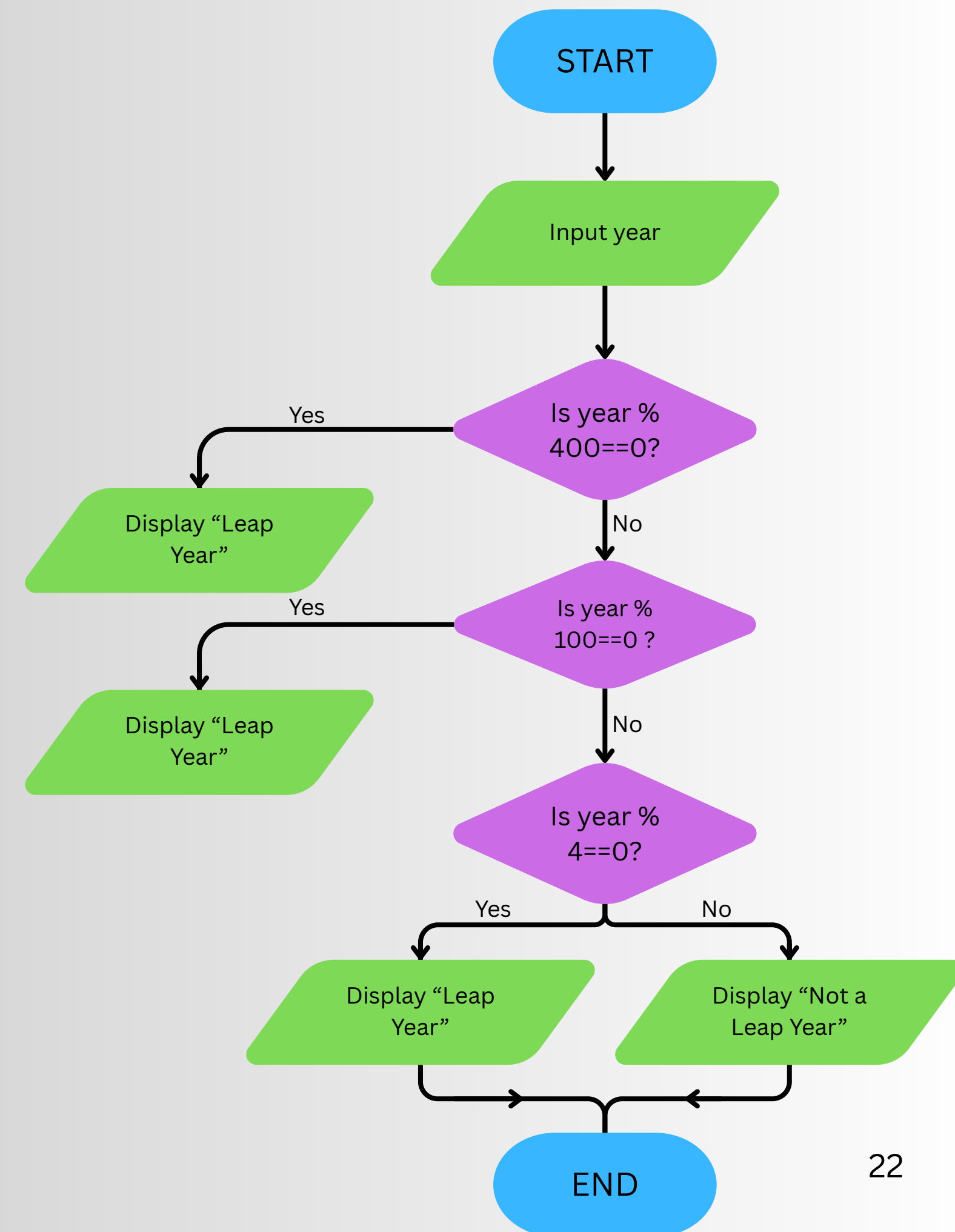
## Q22. Find the maximum among 3 numbers

Algorithm:-

1. Start
2. Declare three integer variables a, b, and c.
3. Prompt: "Enter three numbers:", read a, b, and c.
4. If a > b and a > c, display a is Maximum.
5. Else if b > a and b > c, display b is Maximum.
6. Else, display c is Maximum.
7. End



23

## Q23. Check whether a word is vowel or consonant

Algorithm:-

1. Start
2. Declare a character variable a.
3. Prompt: "Enter a character:", read a.
4. If a is 'a', 'e', 'i', 'o', 'u' (or uppercase),display "a is Vowel".
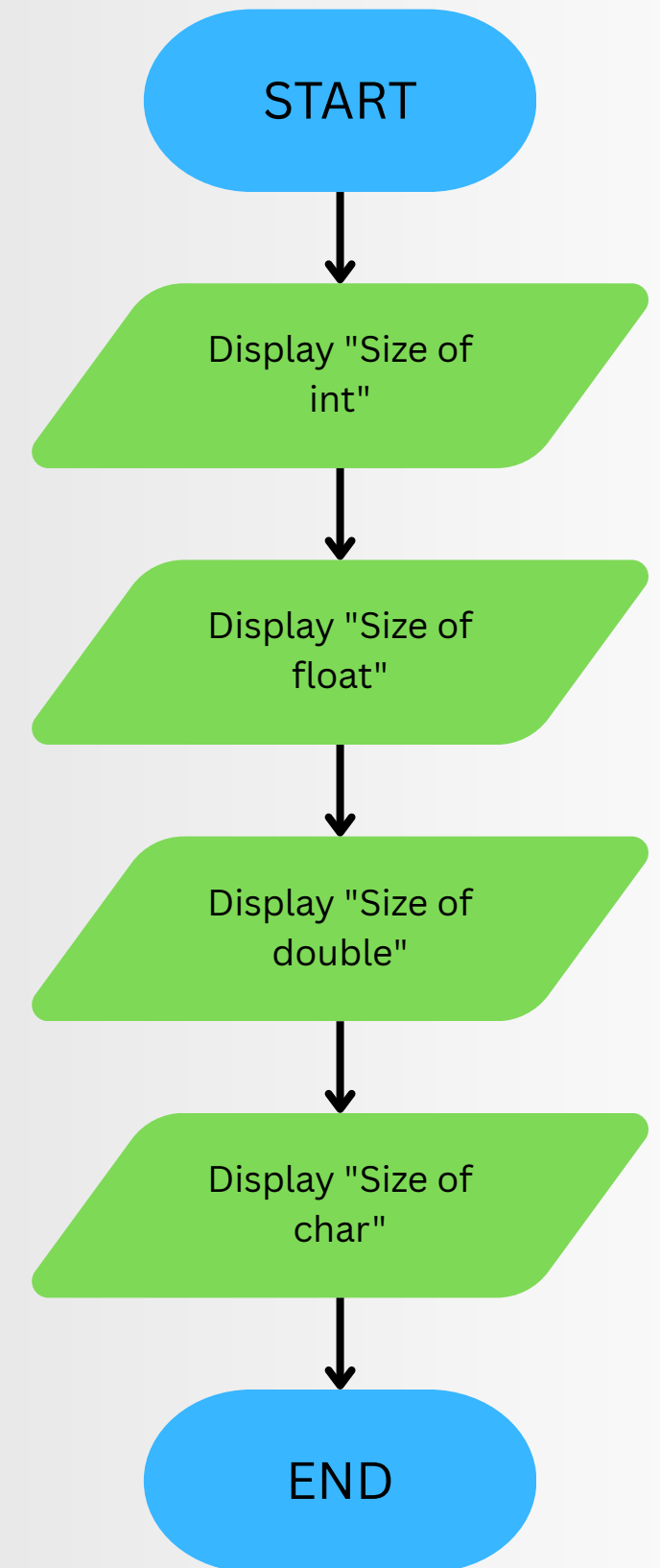5. Else, display "a is a Consonant".
6. End

START

Input character
(a)

Is a == 'a' or 'e' or 'i'
or 'o' or 'u'
or 'A' or 'E' or 'I' or
'O' or 'U'?

Yes                No

Display "a is
Vowel"

Display "a is
Consonant"

END

24

**Q24. Find Sizes of Basic Data Types in C**

Algorithm:-

1. Start
2. Use sizeof operator to determine the size of:
   int,
   float,
   double,
   char
3. Display the sizes of all data types.
4. End



25

## Q25. Check Greatest number among 3 using nested if-else

Algorithm:-

1. Start
2. Declare three integer variables a, b, and c.
3. "Enter three numbers:", read a, b, and c.
4. If a > b
   => If a > c, display a is Greater.
   => Else, display c is Greater.
5. Else
   => If b > c, display b is Greater.
   => Else, display c is Greater.
6. End
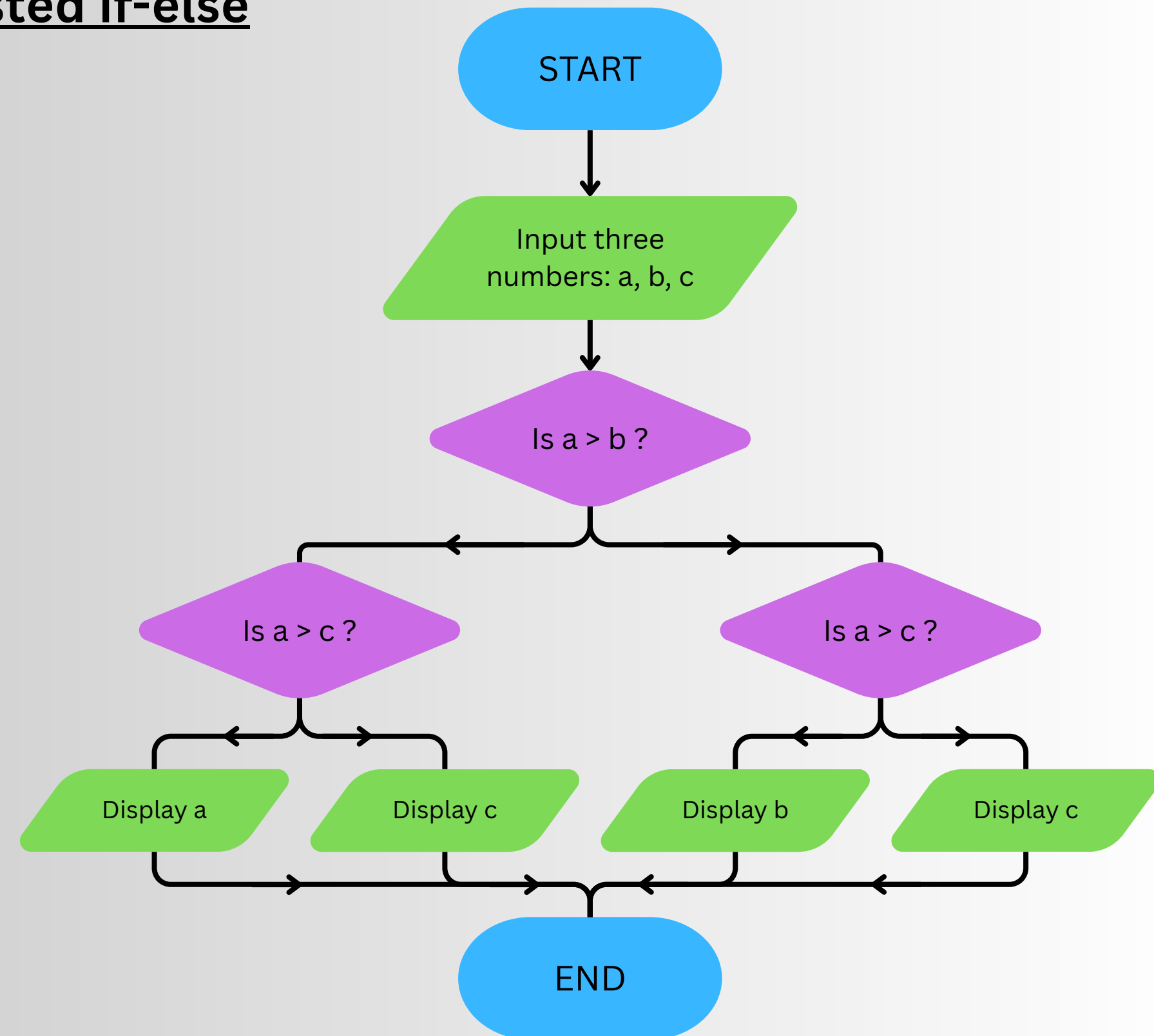
## Q26. Simple Calculator using if-else

Algorithm:-

1. Start
2. Declare variables op (character), n1, n2 (double).
3. Ask the user to enter an operator (+, -, *, /).
4. Read op.
5. Ask the user to enter two numbers.
6. Read n1 and n2.
7. Use switch(op) to perform the operation:
   => If +, display n1 + n2.
   => If -, display n1 - n2.
   => If *, display n1 * n2.
   => If /, check if n2 == 0 → display "Error!", else display n1 / n2.
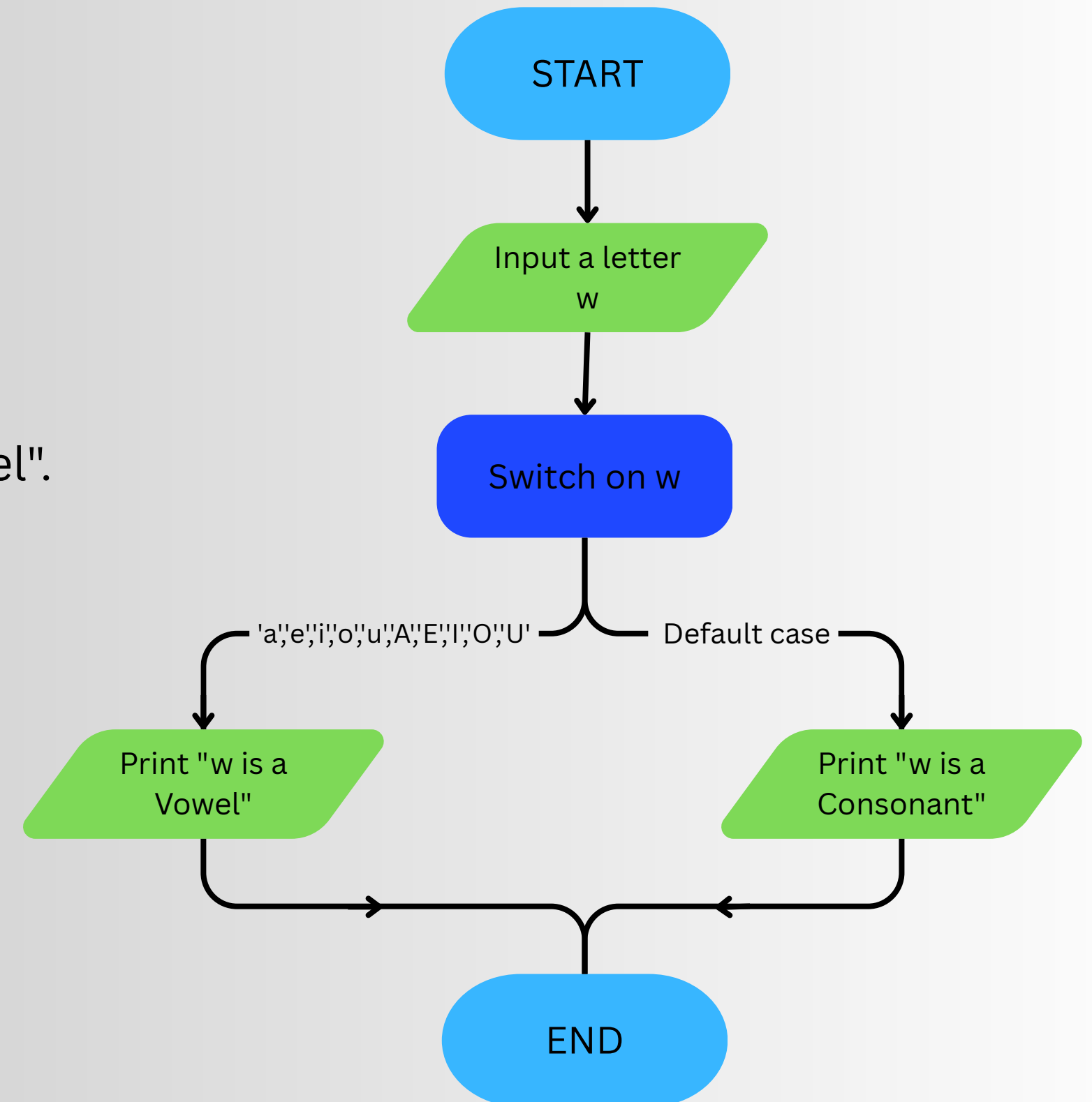   => Otherwise, display "Invalid Input".
8. End

## Flowchart:-
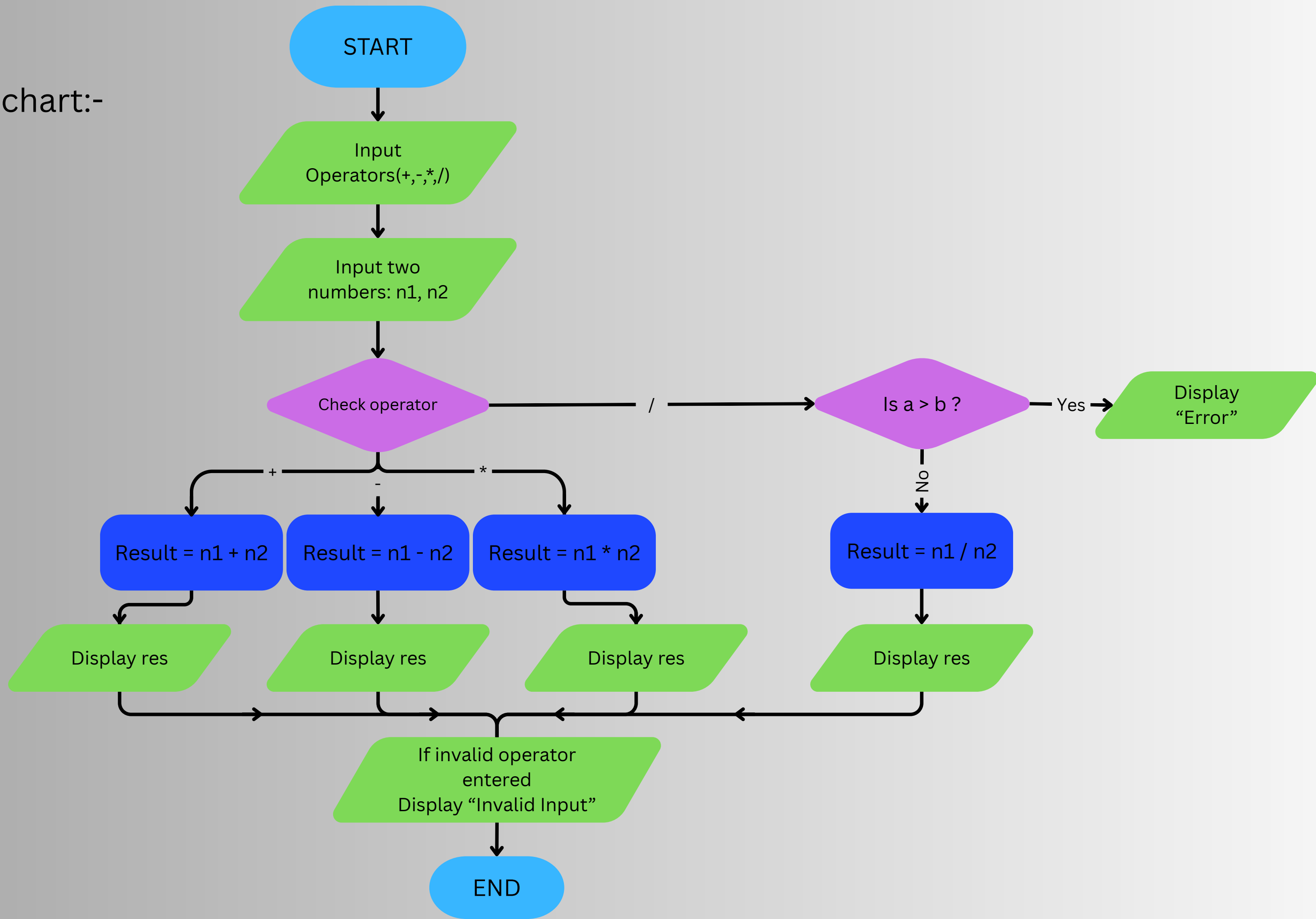
# Q27. Check whether a character is a vowel or consonant using switch case
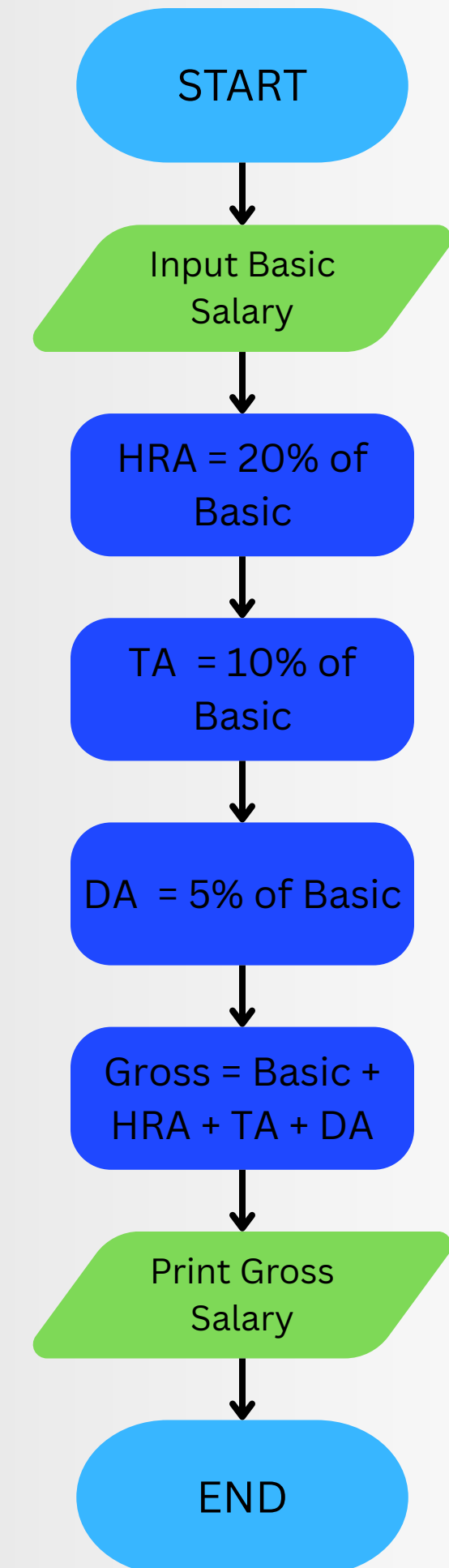
Algorithm:-

1. Start
2. Declare a character variable w.
3. Prompt: "Enter a Letter:", read w.
4. Use switch(w):
5. Case 'a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U' → display "w is a Vowel".
6. Default → display "w is a Consonant".
7. End

## Q28. Calculate Gross Salary

Algorithm:-

1. Start
2. Declare basic, hra, ta, da, and gross as float.
3. Prompt: "Enter Basic Salary:", read basic.
4. Calculate:
   => hra = 20% of basic
   => ta = 10% of basic
   => da = 5% of basic
   => Calculate gross = basic + hra + ta + da.
5. Display "Gross Salary = " followed by gross.
6. End

START

Input Basic Salary

HRA = 20% of Basic

TA = 10% of Basic

DA = 5% of Basic

Gross = Basic + HRA + TA + DA

Print Gross Salary

END

30

## Q29. Calculate Electricity Bill

Algorithm:-

1. Start
2. Declare unit (int) and bill (float).
3. Prompt: "Enter Units:", read unit.
4. If unit <= 100, bill = 5 * unit.
5. Else if unit >= 101 && unit <= 200, bill = 7 * unit.
6. Else, bill = 10 * unit.
7. Display "Electricity Bill = " followed by bill.
8. End
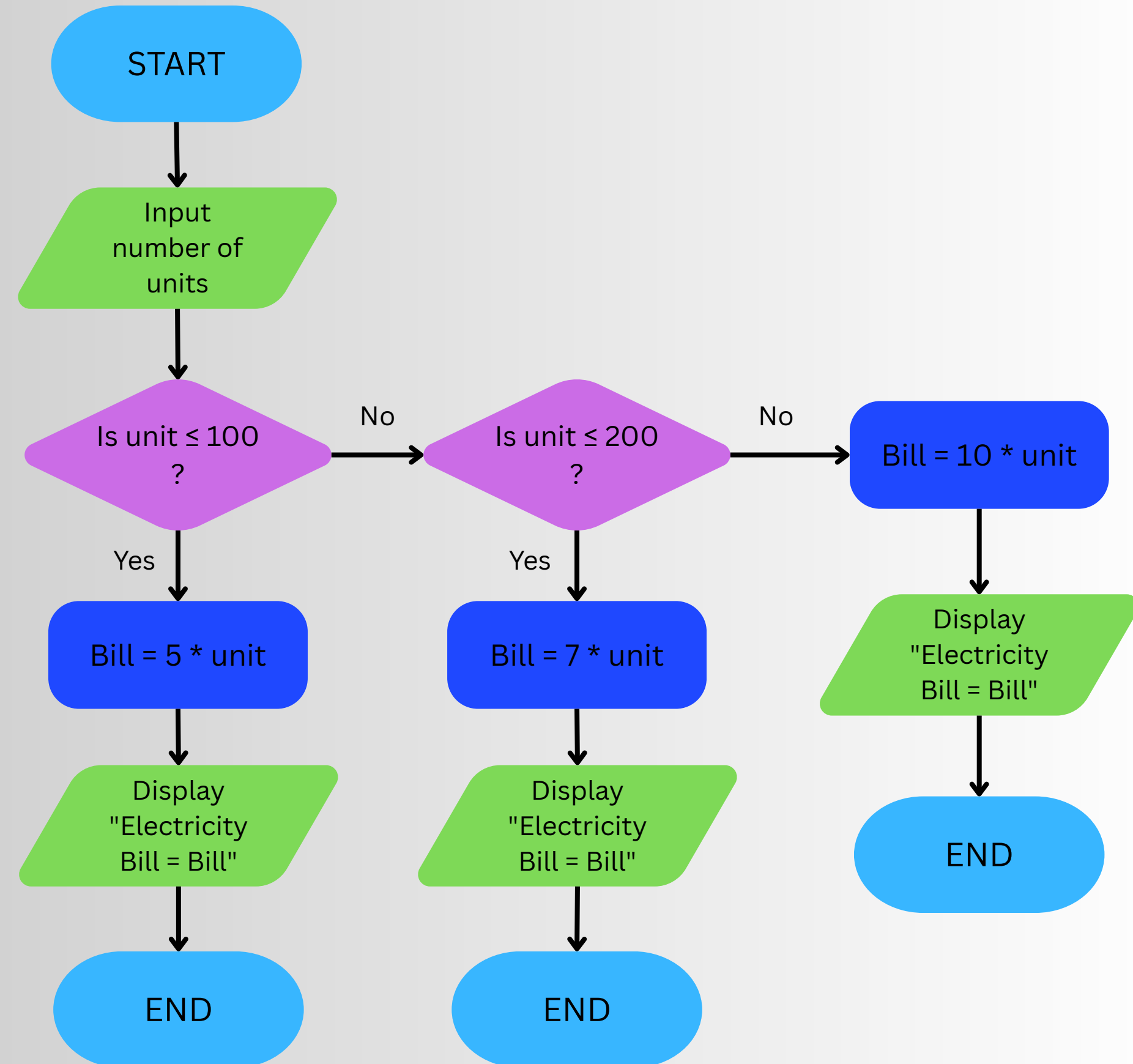
## Q30. Grade Calculation using switch case

Algorithm:-

1. Start
2. Declare marks (int).
3. Prompt: "Enter Marks(out of 100):", read marks.
4. Compute marks / 10 and use switch to decide grade:
   => 10 or 9 → Grade O
   => 8 → Grade A+
   => 7 → Grade A
   => 6 → Grade B+
   => 5 → Grade B
   => 4 → Grade C
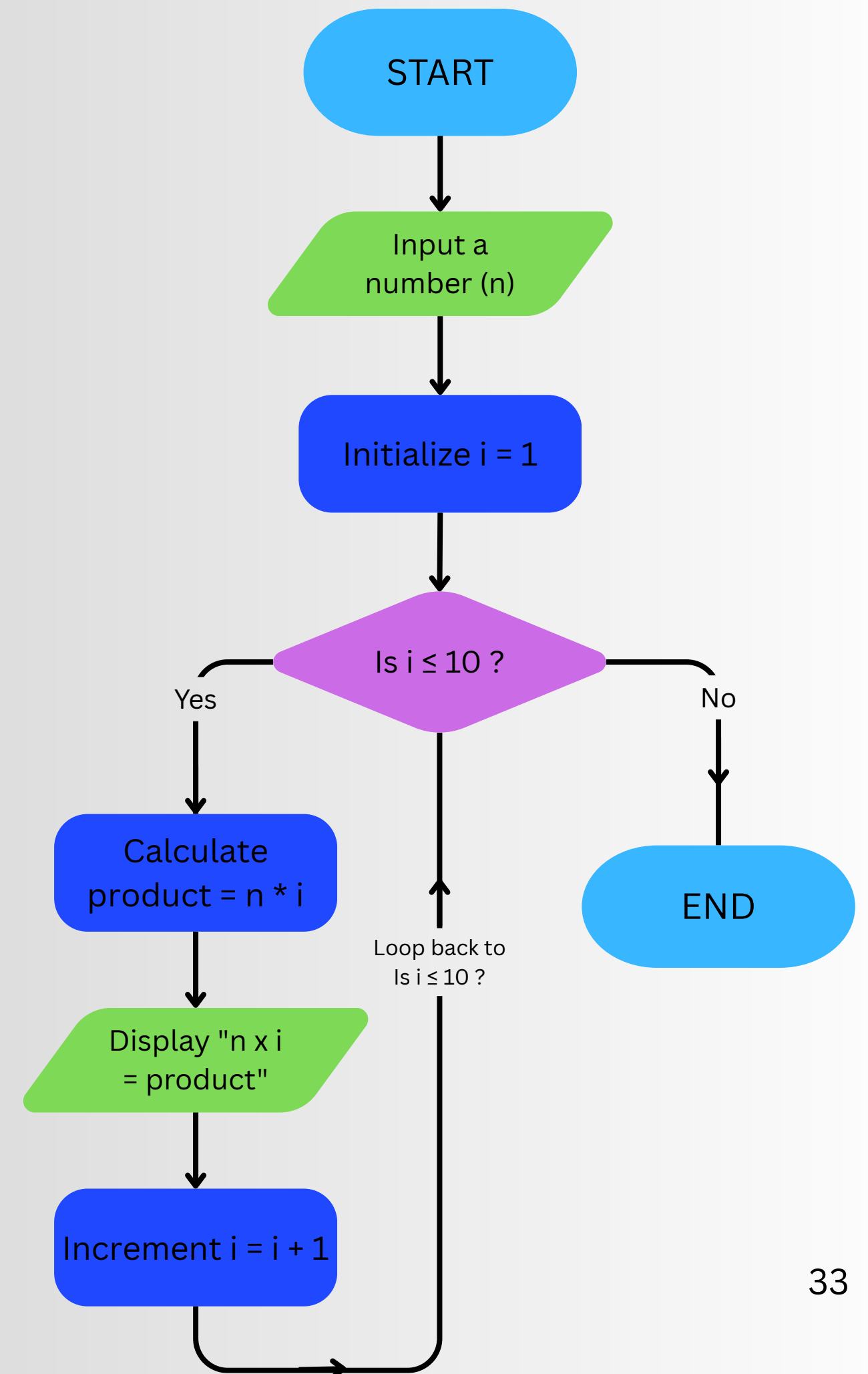   => Default → Grade Fail
5. Display the grade.
6. End

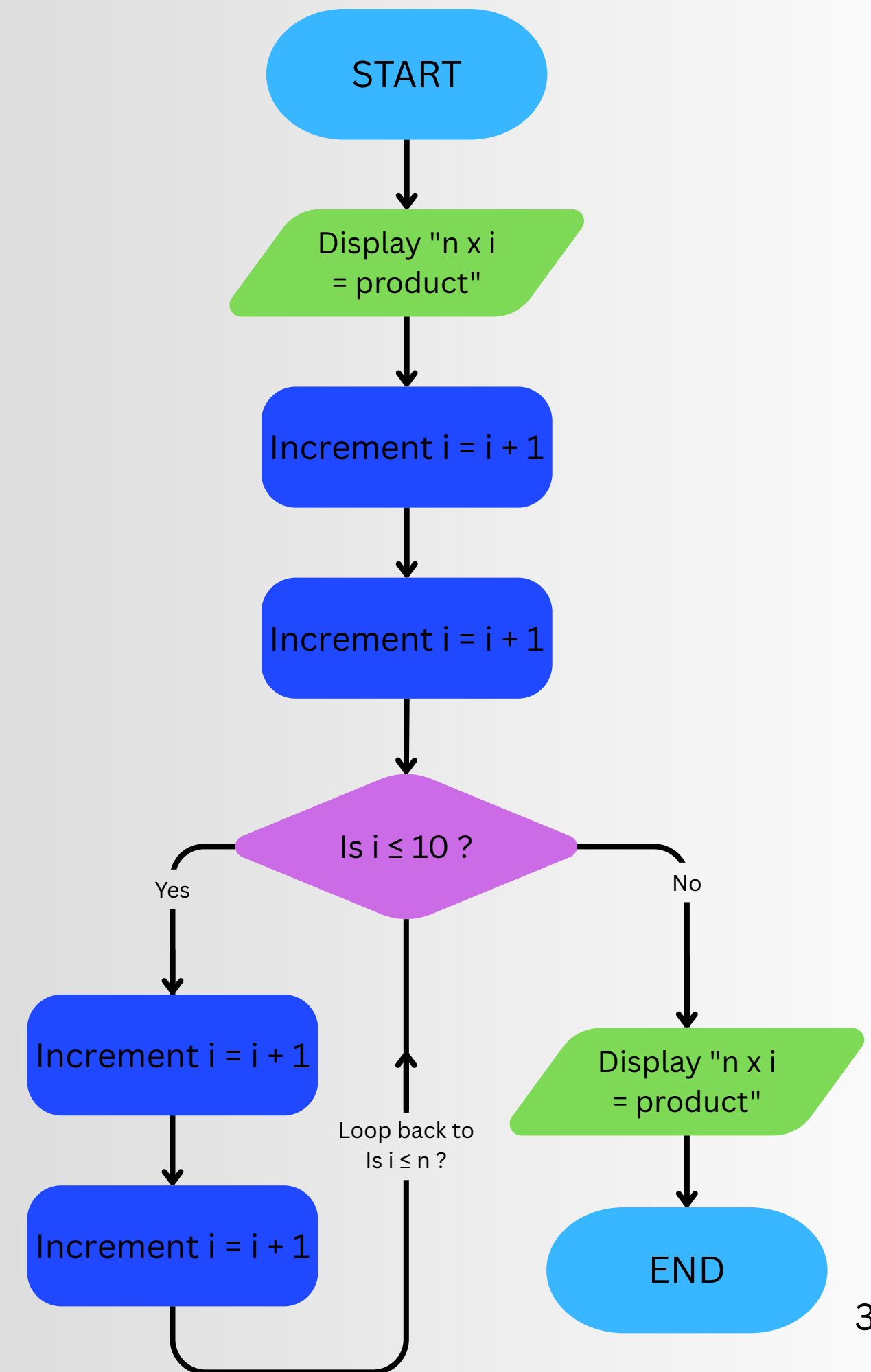

32
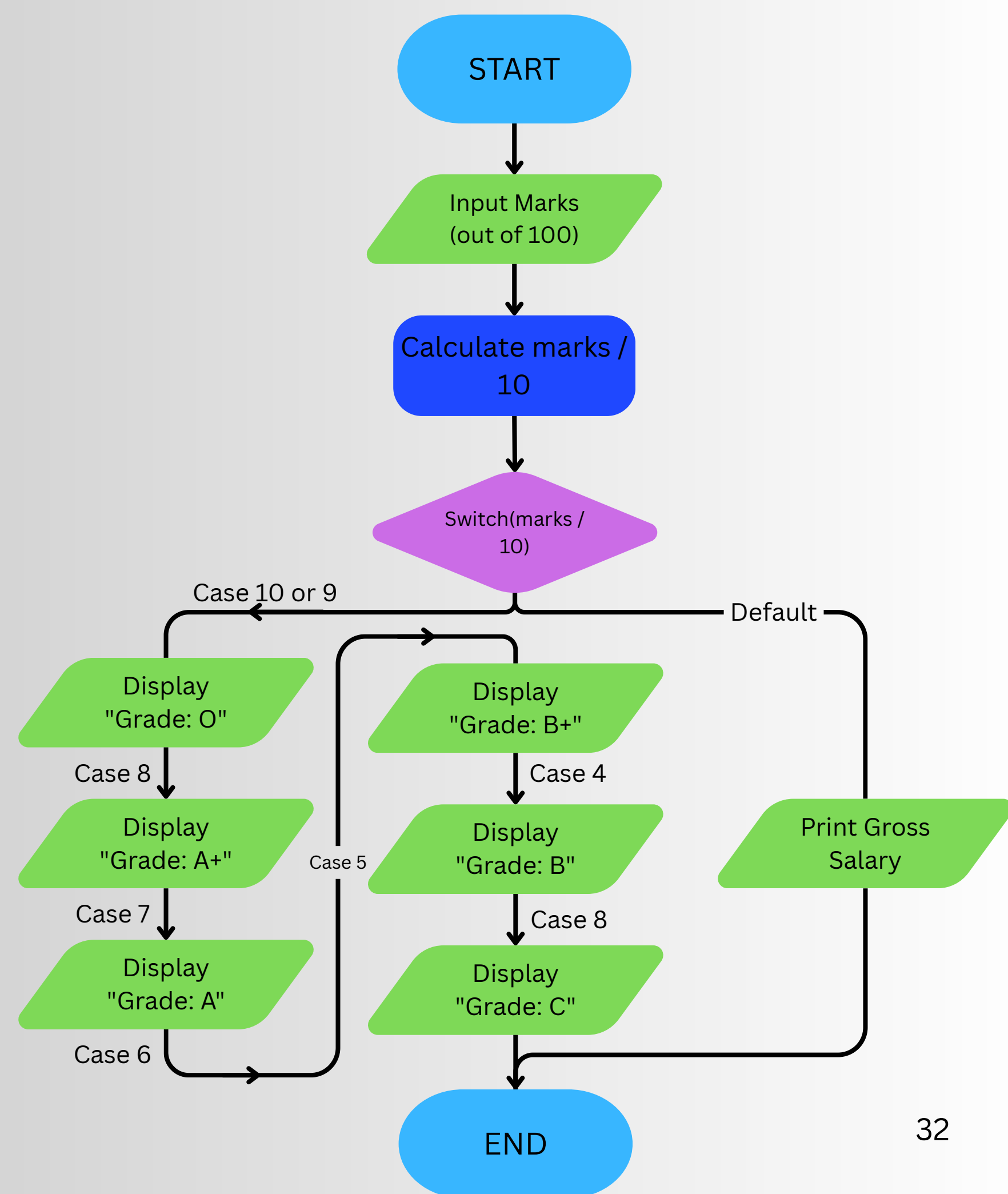
## Q31. Multiplication Table of a number

Algorithm:-

1. Start
2. Declare variable n.
3. Prompt: "Enter Any Number:", then read n.
4. Repeat for i = 1 to 10:
5. Calculate n × i.
6. Display the result in the format n x i = n*i.
7. End



33

**Q32. Sum of First n natural number**

Algorithm:-
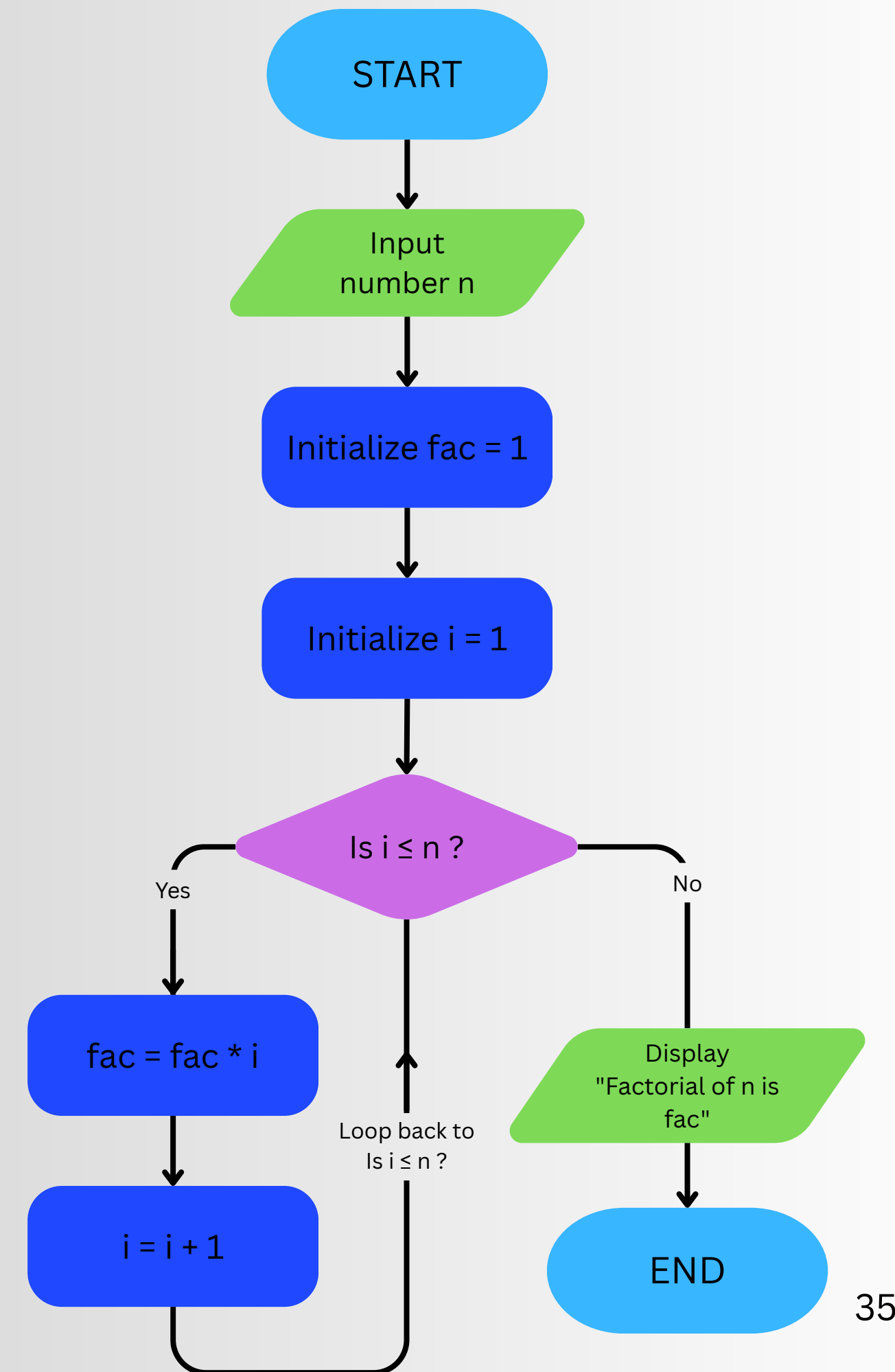
1. Start
2. Read n
3. Set sum = 0
4. Repeat from i = 0 to n: sum = sum + i
5. Print sum
6. End



34

**Q33. Factorial of a number using For loops**

Algorithm:-

1. Start
2. Input number n
3. Set fac = 1
4. Repeat from i = 1 to n: fac = fac * i
5. Display fac
6. End

START

Input number n

Initialize fac = 1

Initialize i = 1

Is i ≤ n ?

Yes

No

fac = fac * i

i = i + 1

Loop back to Is i ≤ n ?

Display "Factorial of n is fac"

END

35

## Q34. Display Fibonacci Series and Its Sum

Algorithm:-

1. Start
2. Declare integer variables n, a = 0, b = 1, s = 0, x = 0.
3. Prompt the user: "Enter Any Number:" and read n.
4. Initialize counter i = 1.
5. While i ≤ n, do the following:
   => Print a.
   => Add a to sum: s = s + a.
   => Compute next term: x = a + b.
   => Update a = b, b = x.
   => Increment i.
6. Display "Sum = " followed by s.
7. End
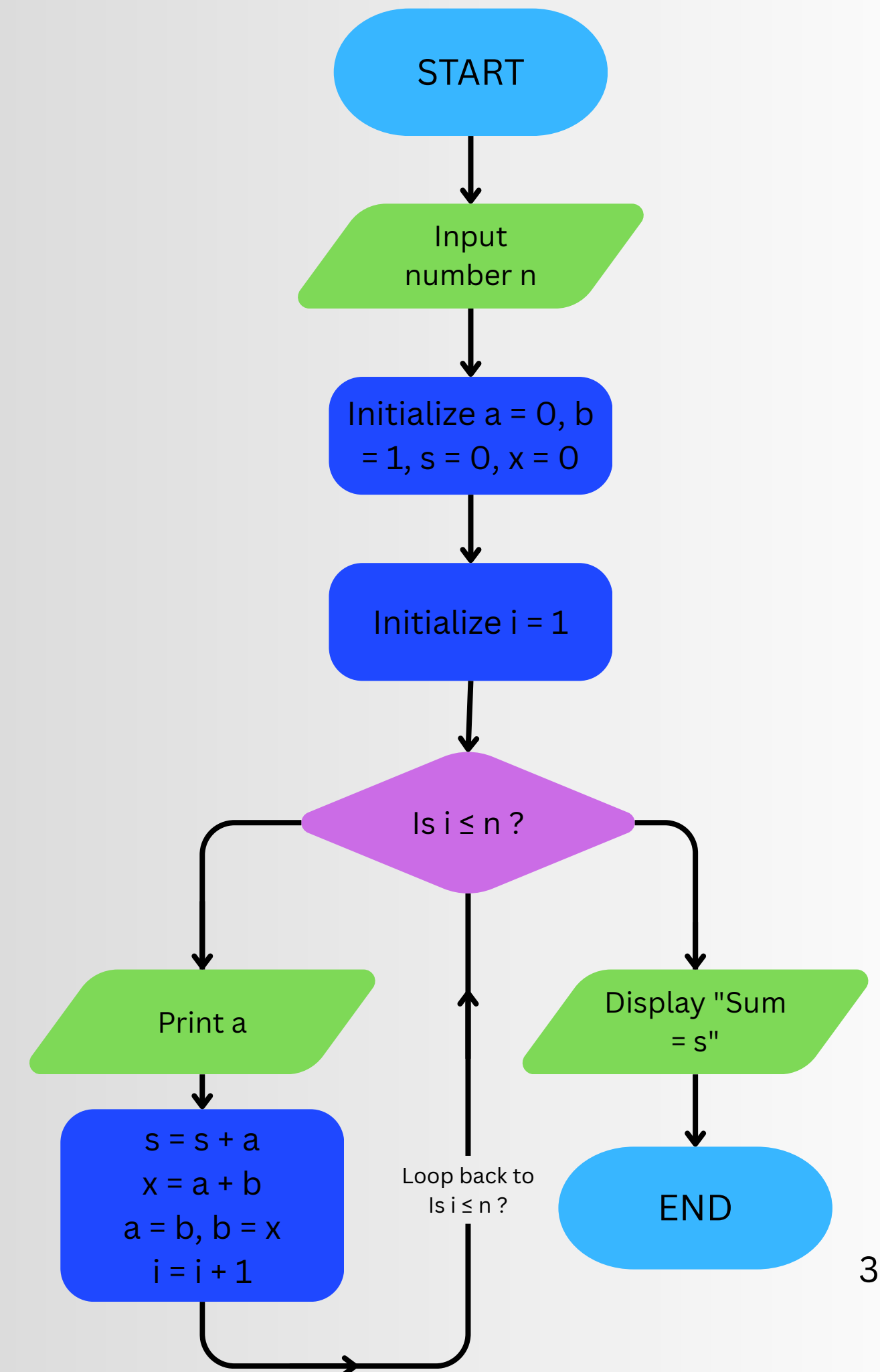
**Q35. Display Prime numbers in a range**

Algorithm:-

1. Start
2. Declare integers st (start) and en (end).
3. Prompt: "Enter the Starting Limit:" and read st.
4. Prompt: "Enter the Ending Limit:" and read en.
5. For i = st to en:
6. Initialize c = 0
7. If i < 2, skip to next iteration (continue)
8. For j = 1 to i:
9. If i % j == 0, increment c by 1
10. If c == 2, print i (prime number)
11. End