

Written By: Javaid Iqbal

### Practicle 1:

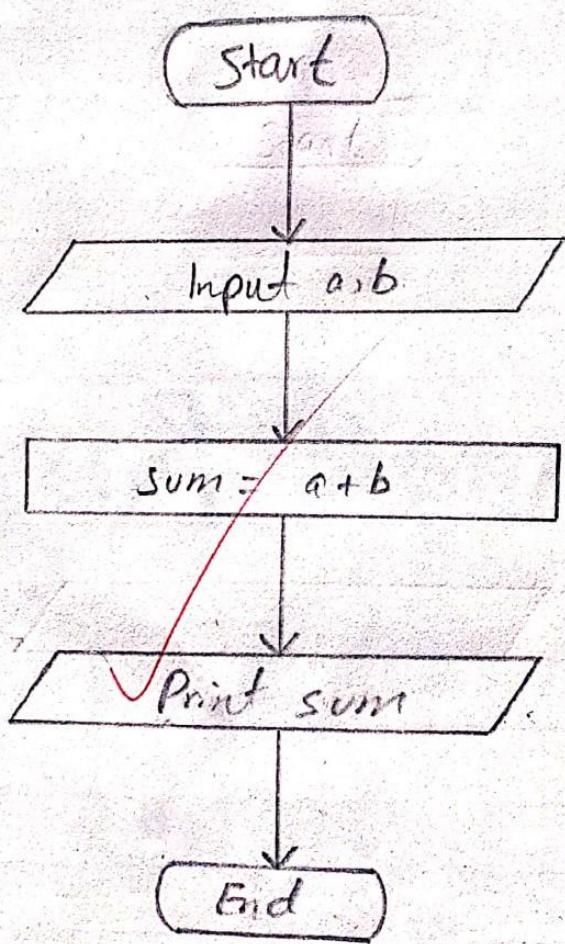
Inputs two numbers, calculates sum and then displays the result on screen.

### Algorithm:

- 1- Start
- 2- Input A
- 3- Input B
- 4- Total = A+B
- 5- Display Total
- 6- Exit

### Program:

```
Private Sub Command1_Click()  
    Dim a, b, sum As Integer  
    a = InputBox("Enter first number:")  
    b = InputBox("Enter Second number")  
    sum = a + b  
    MsgBox "Sum is " & sum  
End Sub
```



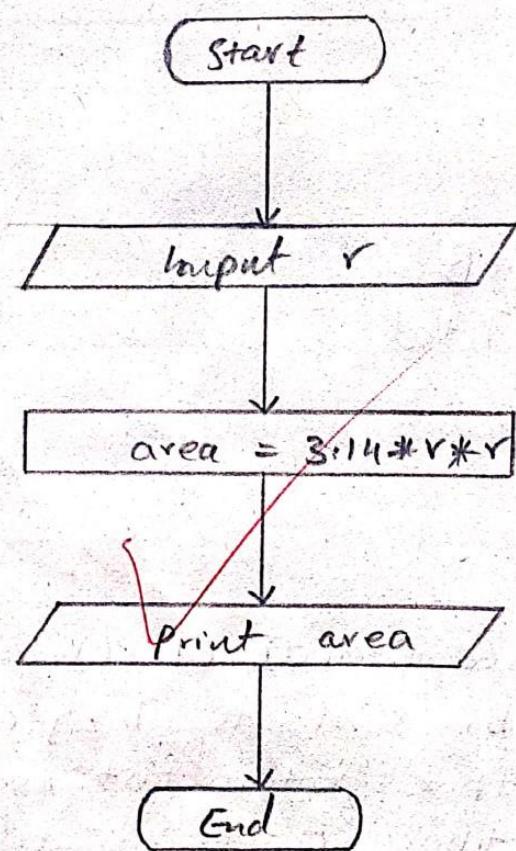
Practice 2:  
Calculate area of circle when  
radius is given:

### Algorithm:

- 1- Start
- 2- Input radius in  $r$
- 3-  $\text{area} = 3.14 * r * r$
- 4- Print area
- 5- End

### Program:

```
Private Sub Command1_Click()  
    Dim r, area As Single  
    r = InputBox("Enter radius:")  
    area = 3.14 * r * r  
    Print area  
End Sub
```



### Practicle: 3

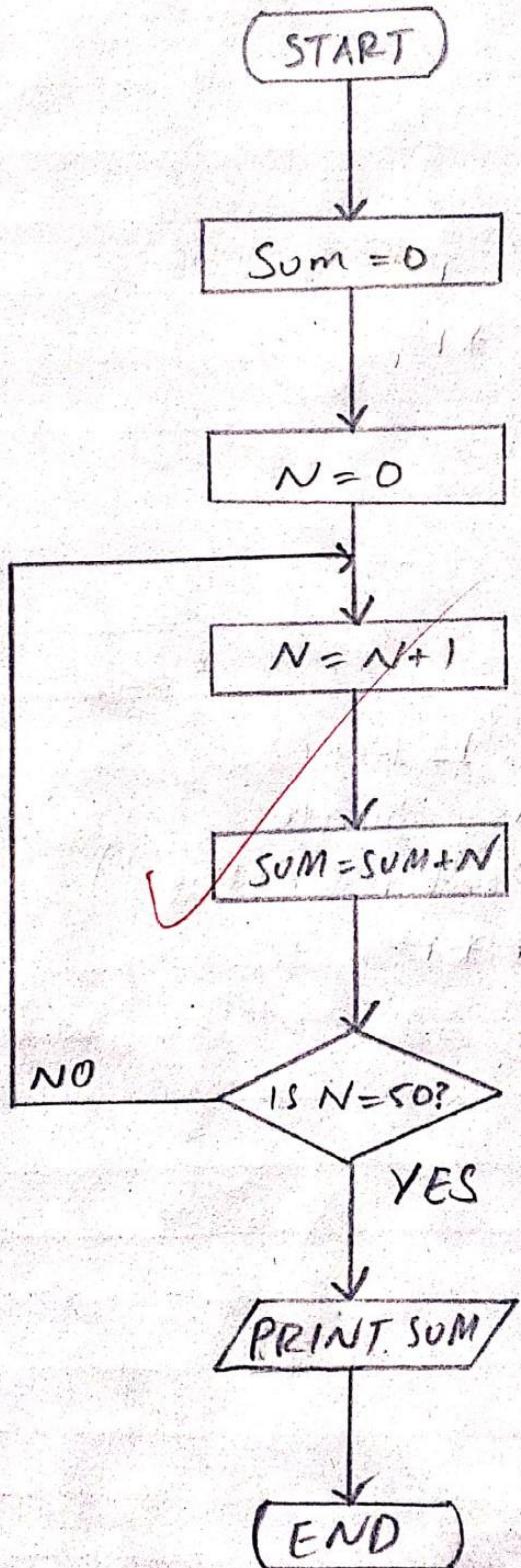
Find the sum of first fifty natural numbers.

### Algorithm:

- 1- Start
- 2- sum = 0
- 3- N=0
- 4- Repeat Step 5 and 6 While ( $N \leq 50$ )
- 5- sum = sum + N
- 6- N= N + 1
- 7- Print sum
- 8- End

### Program:

```
Private Sub Command1_Click()
    Dim i, sum As Integer
    sum = 0
    For i = 1 To 50
        sum = sum + i
        i = i + 1
    Next
    Print "Sum is " & sum
End Sub
```



#### Practice 4:

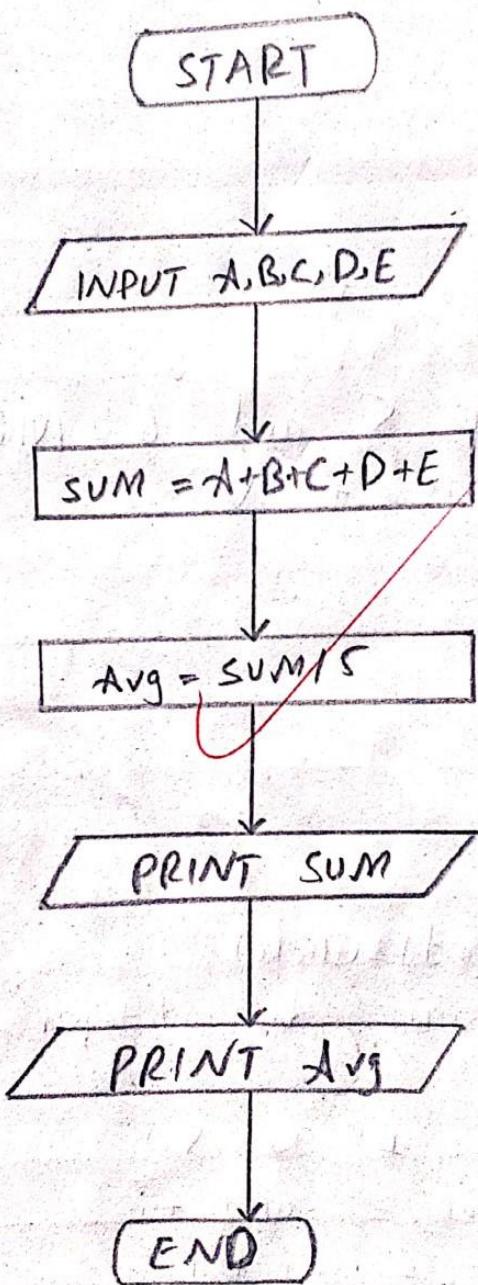
Find the sum and average of five numbers given by the user.

#### Algorithm/Pseudo Code:

- 1- Start
- 2- Input A, B, C, D, E
- 3- Sum = A + B + C + D + E
- 4- Avg = Sum / 5
- 5- Print Sum
- 6- Print Avg
- 7- End

#### Program:

```
Private Sub Command1_Click()  
    Dim a, b, c, d, e, sum As Integer  
    a = Val(InputBox("Enter First Number:"))  
    b = Val(InputBox("Enter Second Number:"))  
    c = Val(InputBox("Enter Third Number:"))  
    d = Val(InputBox("Enter Fourth Number:"))  
    e = Val(InputBox("Enter Fifth Number:"))  
    sum = a + b + c + d + e  
    Print "Sum of given numbers is :" & sum  
End Sub → Print Average of given num is :" & sum / 5
```



## Practical 5:

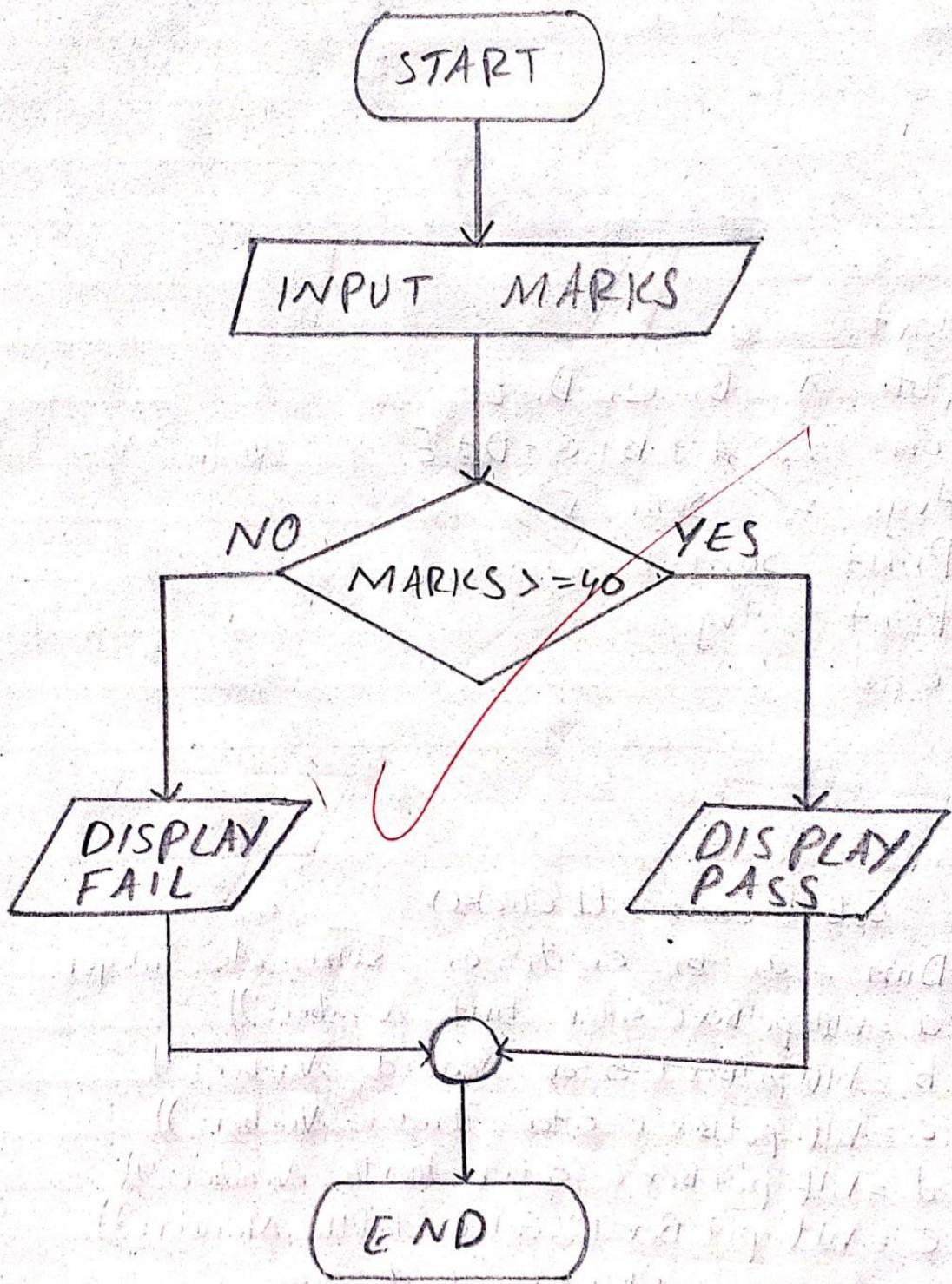
Inputs the marks of the student.  
It displays "Pass" if marks  
are  $\geq 40$  or more. Otherwise display  
"Fail".

## Pseudo Code:

- 1- Start
- 2- Input marks
- 3- If marks  $\geq 40$  Then
- 4- Display Pass
- 5- Else
- 6- Display Fail
- 7- End

## Program:

```
Private Sub Command1_Click()
    Dim m As Integer
    m = InputBox("Enter Marks:")
    If m  $\geq 40$  Then
        MsgBox "Pass"
    Else
        MsgBox "Fail"
    End If
End Sub
```

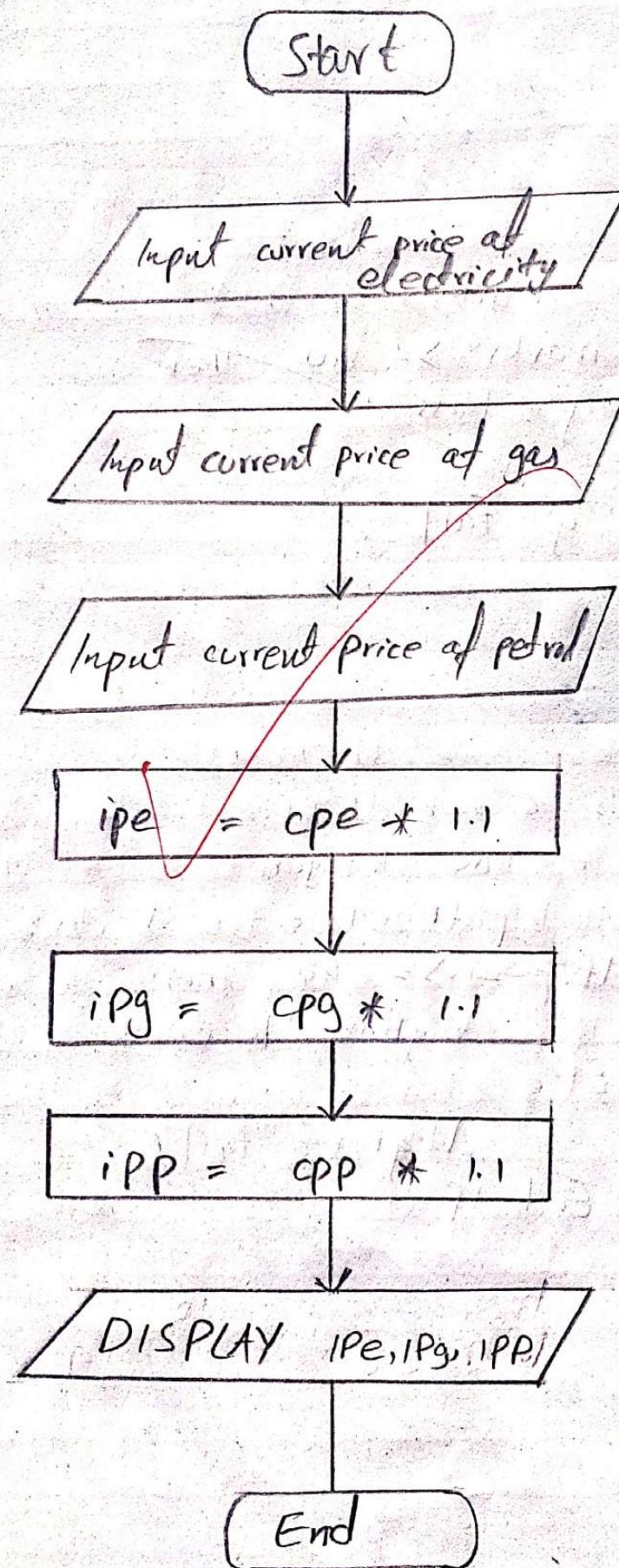


## Practical 6:

Write a program that will prompt the user to enter the current rates of electricity, gas and petrol per unit. Give each item's rate an increment of 10 %. Compute and displays the new prices per unit of electricity, gas and petrol.

### Program:

```
Private Sub Command1_Click()
Dim cpe, cpq, cpp As Single
Dim ipe, ipg, ipp As Single
cpe = InputBox ("Enter current price of electricity:")
cpq = InputBox ("Enter current price of gas:")
cpp = InputBox ("Enter current price of petrol:")
ipe = cpe * 1.1
ipg = cpq * 1.1
ipp = cpp * 1.1
MsgBox ("New prices per unit:" & vbCrLf
        & "Electricity:" & ipe & vbCrLf
        & "Gas:" & ipg & vbCrLf
        & "Petrol:" & ipp)
End Sub
```

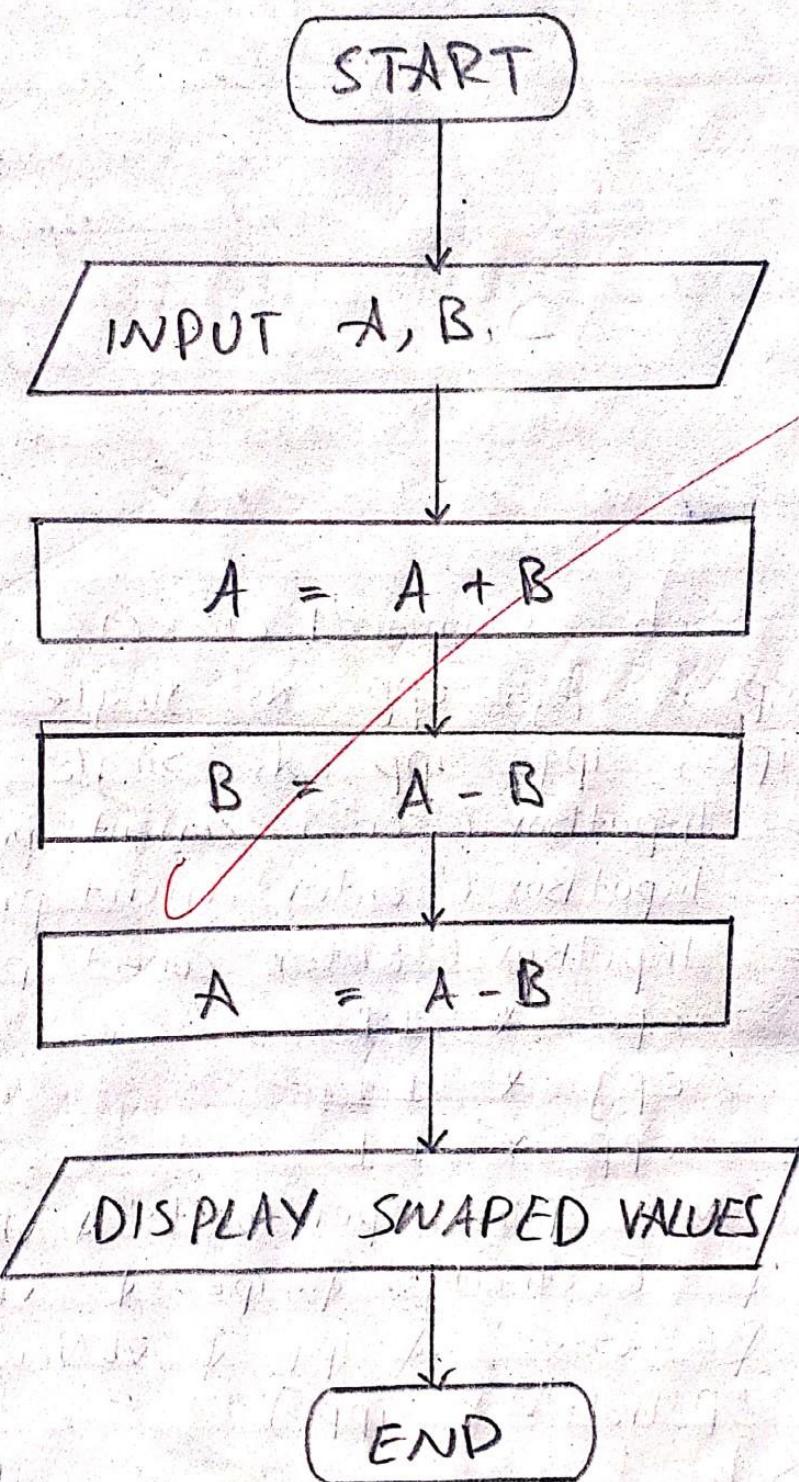


### Practice 7:

Write a program that inputs two numbers and swaps those numbers without using third variable.

### Program:

```
Private Sub Command1_Click()
    Dim A, B As Integer
    A = InputBox("Enter First Number:")
    B = InputBox("Enter Second Number:")
    Print "Older Values:" & vbCrLf & "A=" & A, "B=" & B
    A = A + B
    B = A - B
    A = A - B
    Print "New Values:" & vbCrLf & "A=" & A, "B=" & B
End Sub
```

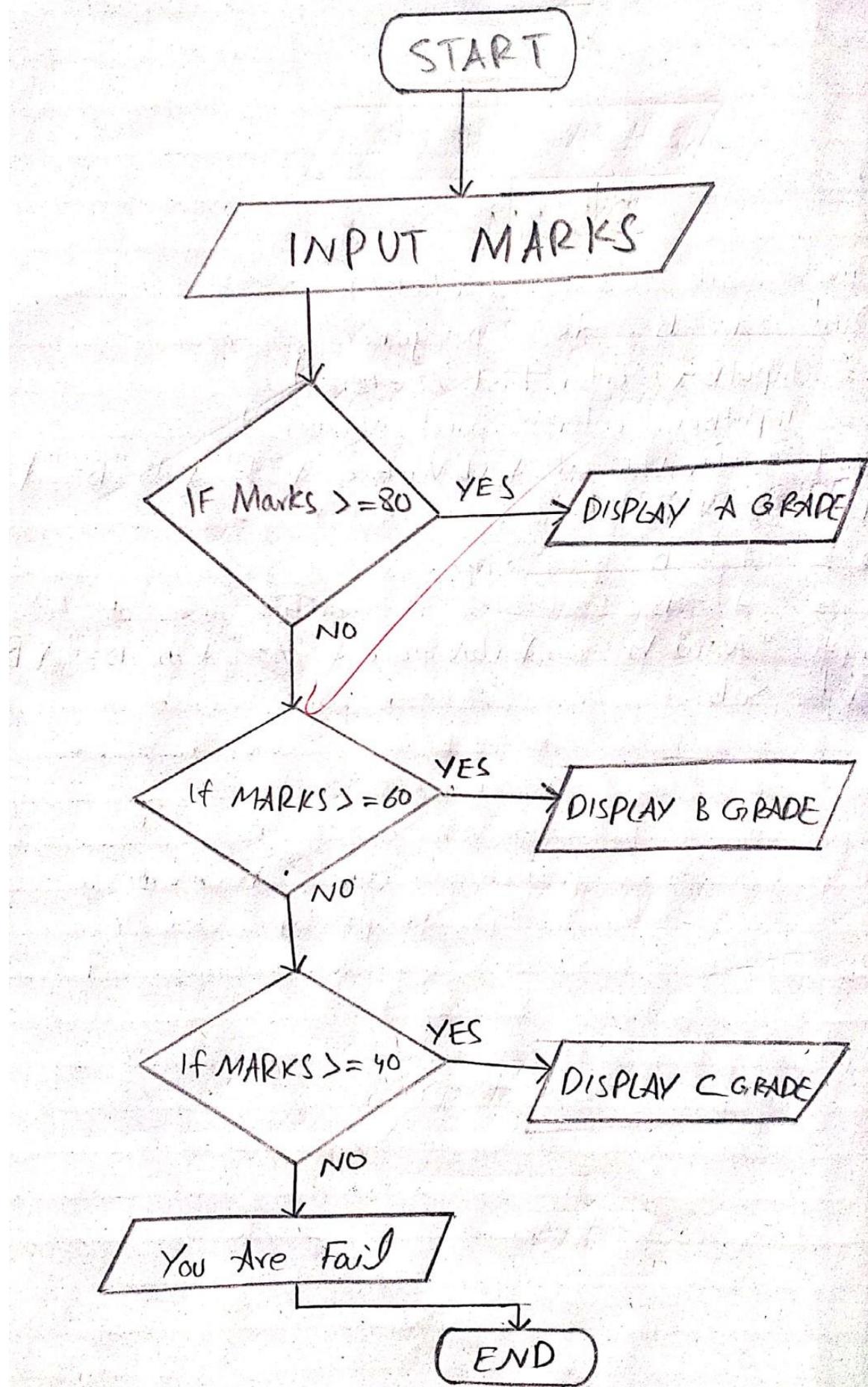


### Practicole 8:

Write a program that inputs marks. It displays "A grade" if marks are 80 or above, "B grade" if marks are b/w 60 & 79, "C grade" if marks are b/w 40 & 59 and otherwise display "you are fail."

### Program:

```
Private Sub Command1_Click()
    Dim marks As Integer
    If marks >= 80 Then
        MsgBox "A grade"
    ElseIf marks >= 60 And marks < 80 Then
        MsgBox "B grade"
    ElseIf marks >= 40 And marks < 60 Then
        MsgBox "C grade"
    Else
        MsgBox "You are fail."
    End If
End Sub
```

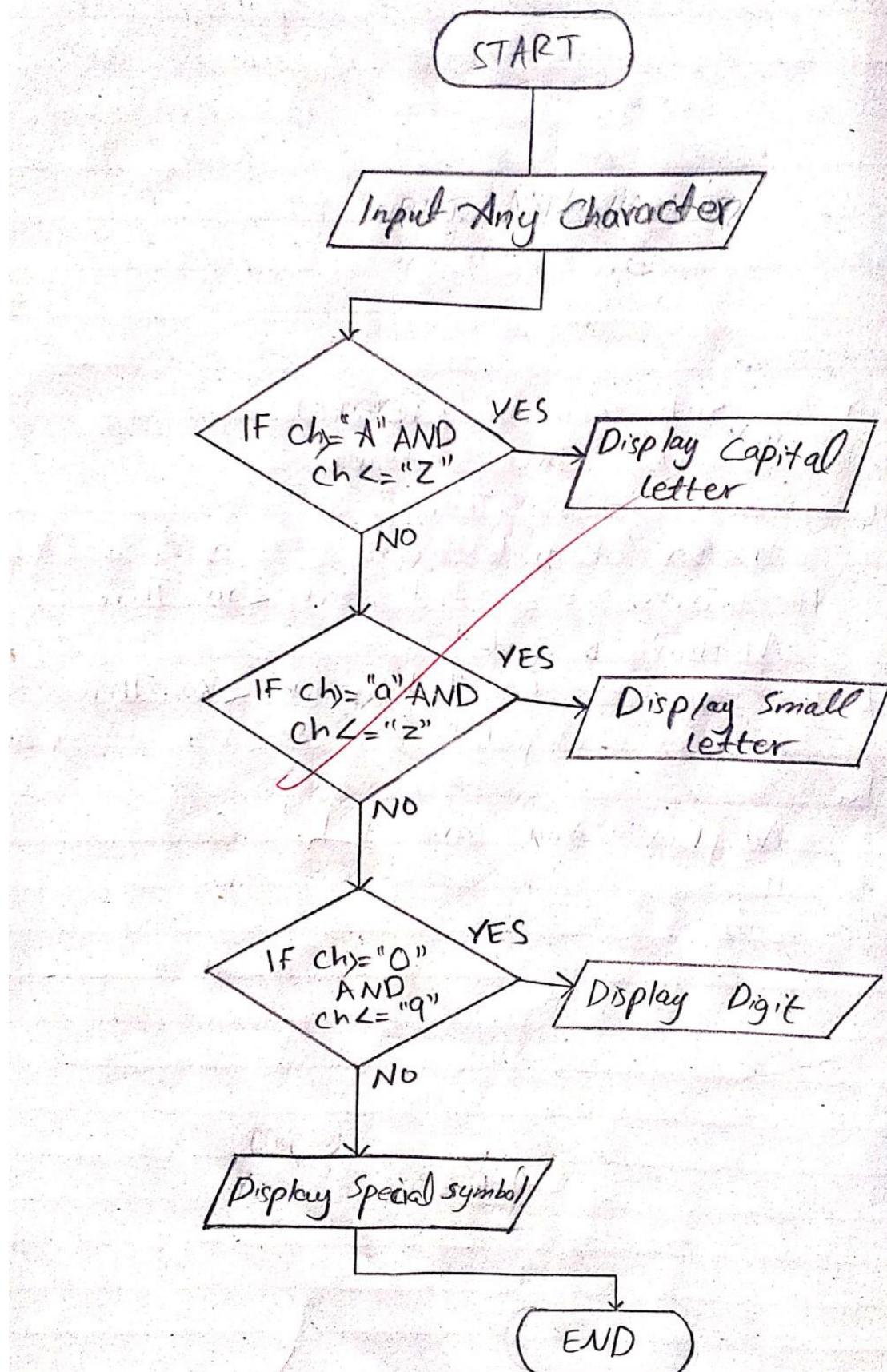


### Practical 9:

Write a program that allows user to enter character through Keyboard and determines whether the character is capital letter, small case letter, a digit or a special symbol.

### Program:

```
Private Sub Command1_Click()
Dim ch As String
ch = InputBox("Enter any character!")
If ch >= "A" And ch <= "Z" Then
    MsgBox "You enter a Capital letter."
ElseIf ch >= "a" And ch <= "z" Then
    MsgBox "You enter a small letter."
ElseIf ch >= "0" And ch <= "9" Then
    MsgBox "You enter a digit."
Else
    MsgBox "You enter a special symbol."
End If
End Sub
```

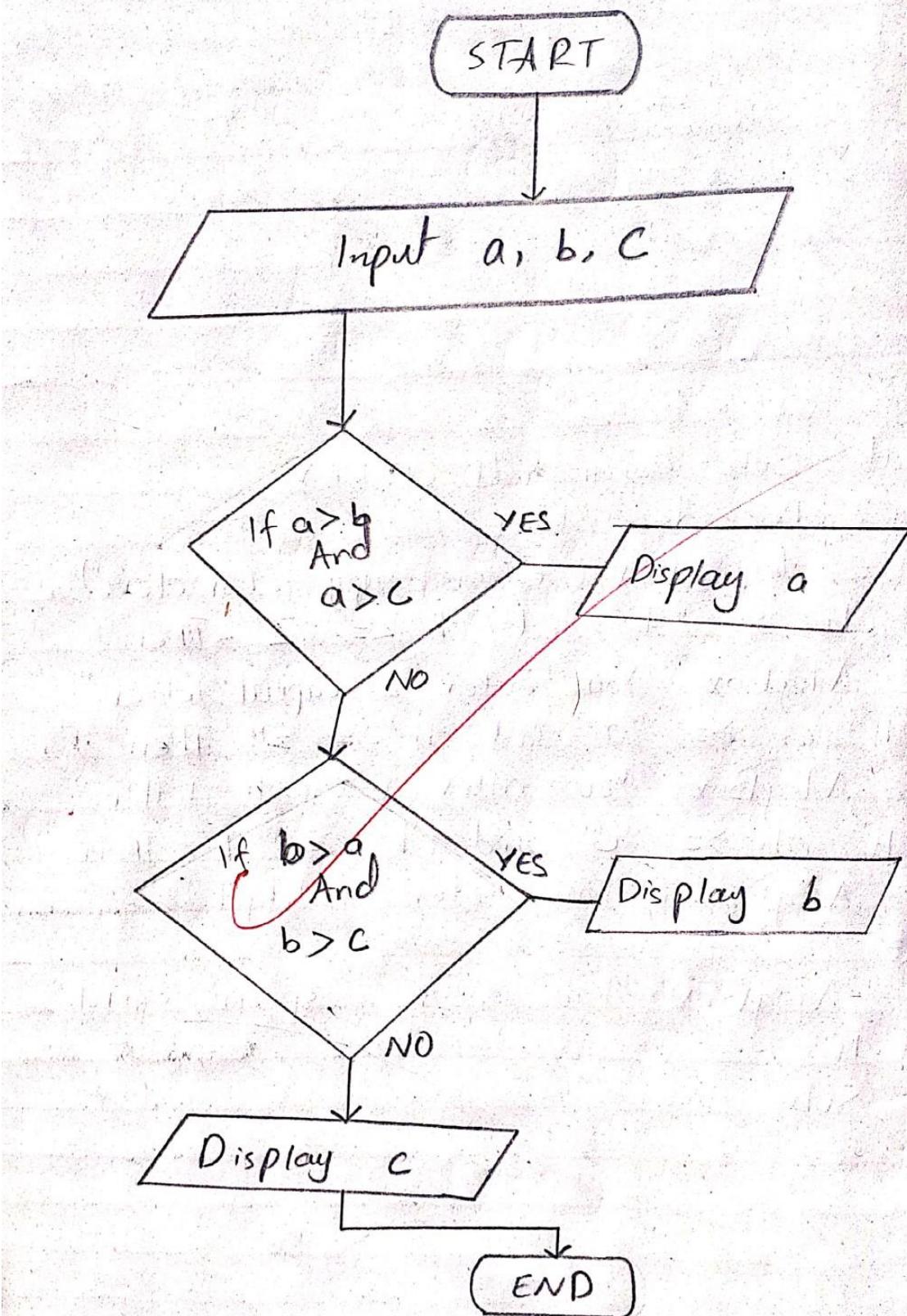


## Practical 10:

Write a program that inputs three numbers and displays the maximum number using logical operation.

### Program:

```
Private Sub Command1_Click()
Dim a, b, c As Integer
a = InputBox ("Enter 1st Number:")
b = InputBox ("Enter 2nd Number:")
c = InputBox ("Enter 3rd Number:")
If a > b And a > c Then
    MsgBox "Maximum number is:" & a
ElseIf b > a And b > c Then
    MsgBox "Maximum number is:" & b
Else
    MsgBox "Maximum number is:" & c
End If
End Sub
```

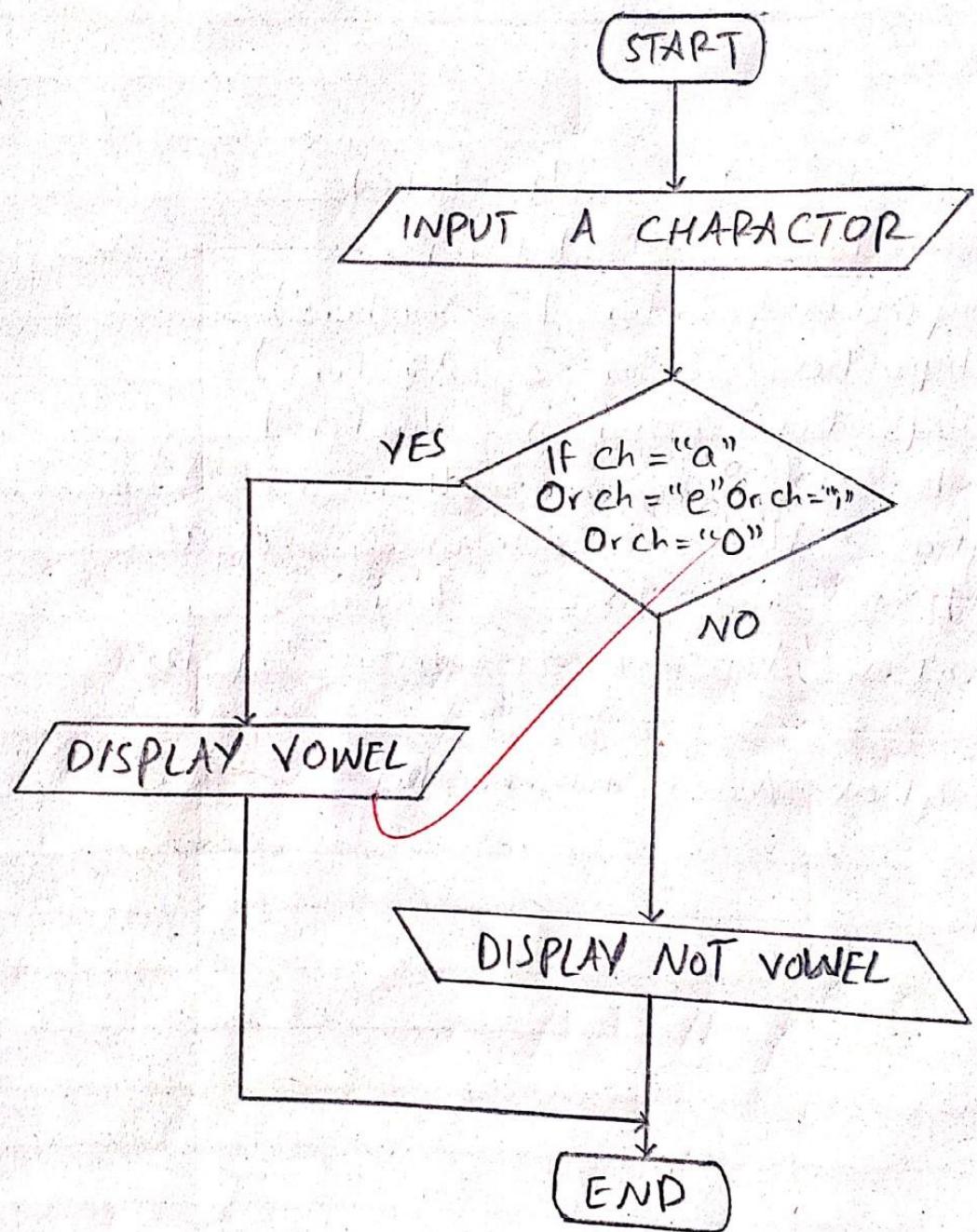


### Practical 11:

Write a program that inputs a character and displays whether it is vowel or not.

### Program:

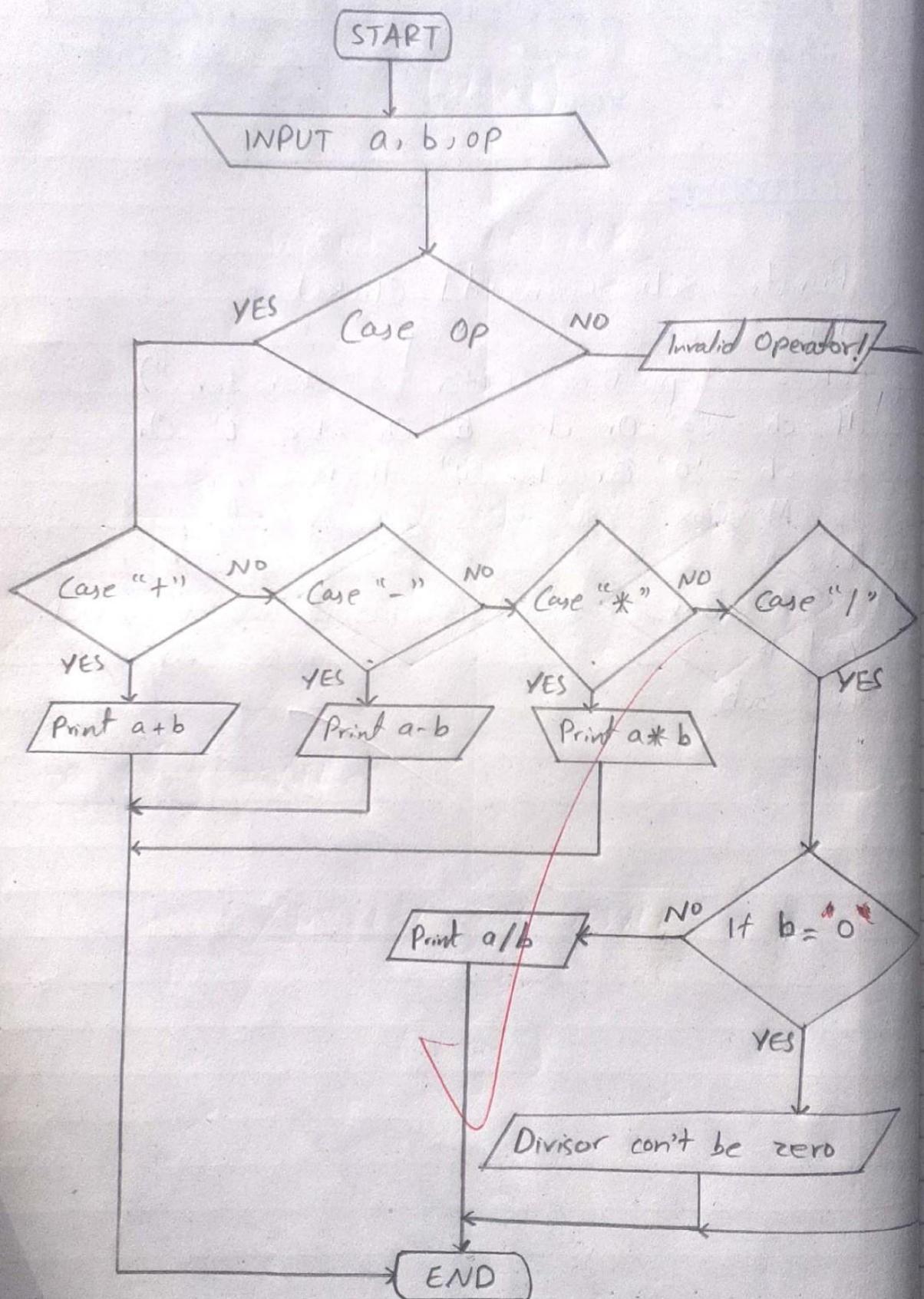
```
Private Sub Command1_Click()
Dim ch As String
ch = InputBox("Enter a character")
If ch = "a" Or ch = "e" Or ch = "i" Or
ch = "o" Or ch = "u" Then
    MsgBox "You enter a Vowel."
Else
    MsgBox "Entered character is not Vowel."
End If
End Sub
```



## Practicile 12:

Write a program that inputs a floating point number, an operator and another floating point number. It displays the result of performing the operation on the given numbers. If the operator is division, it should check to make sure the divisor is not equal to zero. If the operator is not a +, -, \*, or / then the program should print "Invalid operator" message.

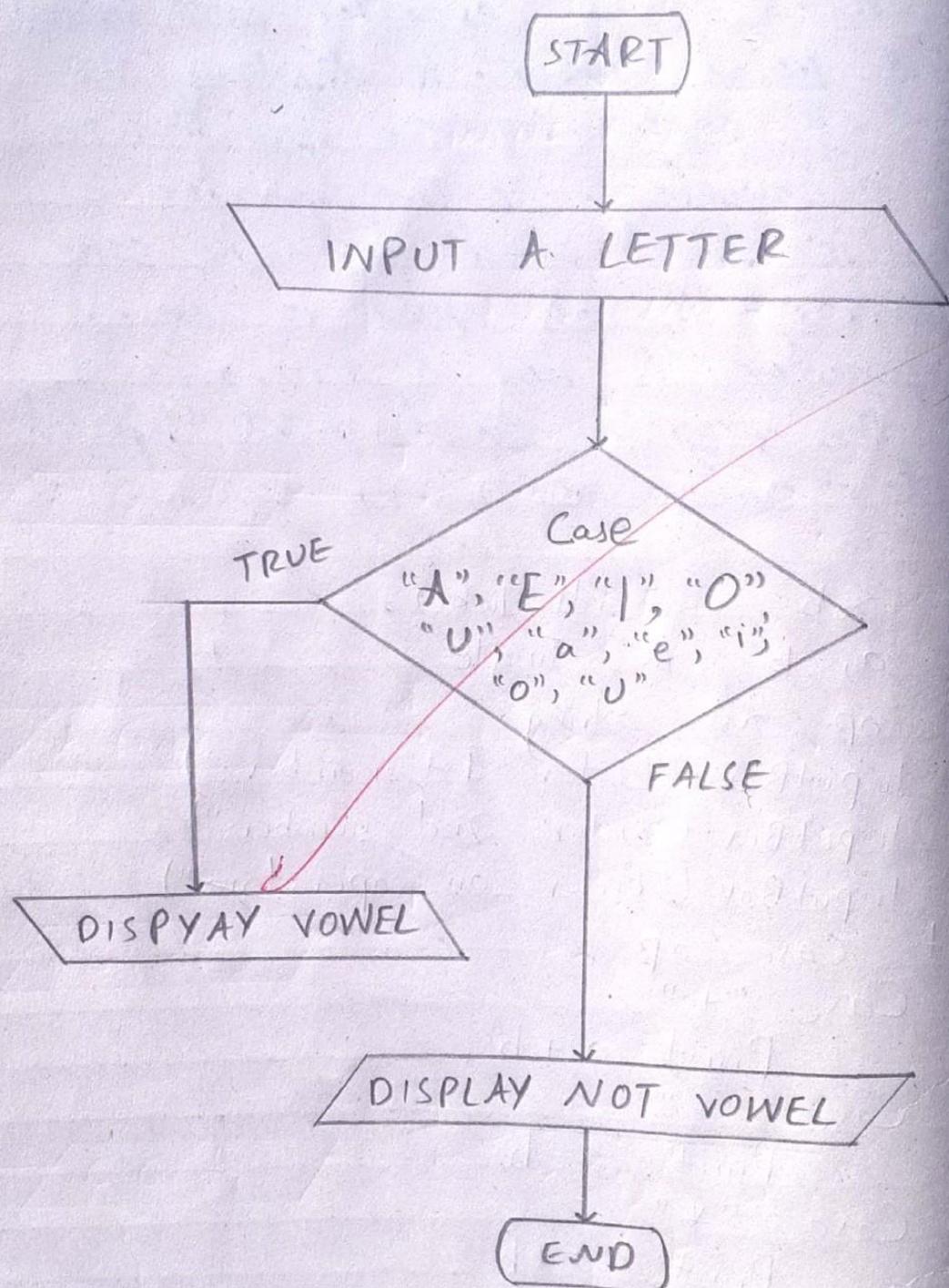
```
Private Sub Command1_Click()
    Dim a, b As Single
    Dim op As String
    a = InputBox("Enter 1st number:")
    b = InputBox("Enter 2nd number:")
    op = InputBox("Enter an operator:")
    Select Case op
        Case "+"
            Print a + b
        Case "-"
            Print a - b
        Case "*"
            Print a * b
        Case "/"
            If b = 0 Then
                MsgBox "Divisor can't be zero."
            Else
                Print a / b
            End If
        Case Else
            MsgBox "Invalid Operator!"
    End Select
End Sub
```



### Practical 13:

Write a program that gets a letter from the user and tells whether it is vowel or not using Select .... Case Structure.

```
Private Sub Command1_Click()
Dim letter As String
letter = InputBox ("Enter a letter")
Select Case letter
Case "A", "E", "I", "O", "U", "a", "e", "i", "o", "u"
    MsgBox "You enter a vowel."
Case Else
    MsgBox "Letter is not vowel."
End Select
End Sub
```

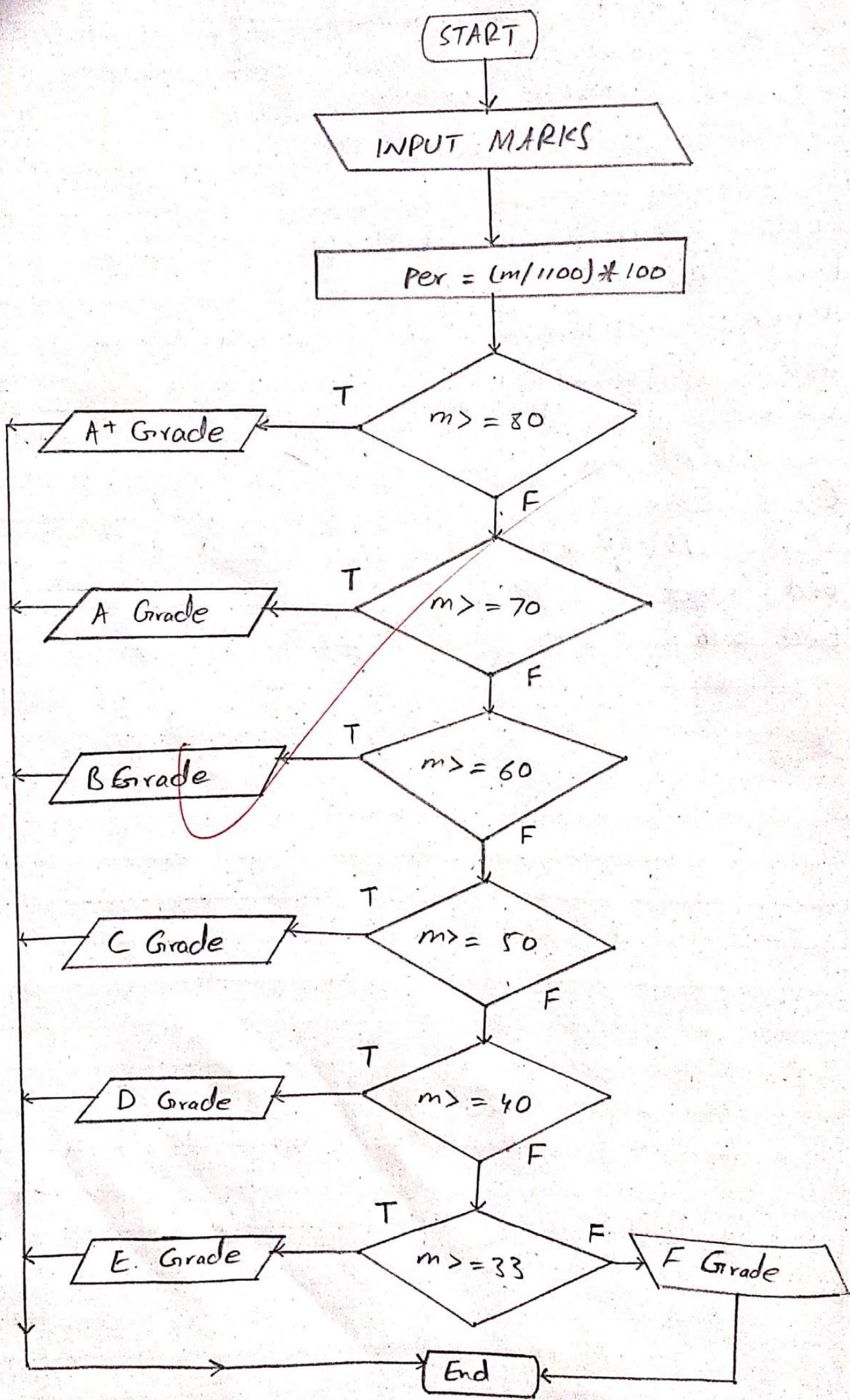


## Practical 14:

Write a program that inputs obtained marks of a student, calculates the percentage and display the grades (assuming total marks as 100).

### Program:

```
Private Sub Command1_Click()
    Dim m, per As Integer
    m = InputBox("Enter your obtained marks:")
    per = (m / 100) * 100
    Select Case per
        Case 80 To 100
            MsgBox "A+ Grade"
        Case 70 To 79
            MsgBox "A Grade"
        Case 60 To 69
            MsgBox "B Grade"
        Case 50 To 59
            MsgBox "C Grade"
        Case 40 To 49
            MsgBox "D Grade"
        Case 33 To 39
            MsgBox "E Grade"
        Case Else
            MsgBox "F Grade"
    End Select
End Sub
```



### Program 15:

Write a program that inputs a number and checks if it is fibonacci number or not. In fibonacci sequence, sum of two successive terms gives the third term.

```
Private Sub Command1_Click()
```

```
Dim a, b, c, n As Integer
```

```
n = InputBox("Enter a number:")
```

```
a = 0
```

```
b = 1
```

```
c = a + b
```

```
While c < n
```

```
    c = a + b
```

```
    a = b
```

```
    b = c
```

```
Wend
```

```
If n = c Then
```

```
    MsgBox ("The number is a fibonacci num.")
```

```
Else
```

```
    MsgBox ("The number is not a fibonacci num.")
```

```
End If
```

```
End Sub
```

## Practical 16:

Write a program that will prompt the user for table number and limit up to which the table should be printed using inputbox function. The table should be printed on the form as per the user inputs.

```
Private Sub Command1_Click()
Dim tn, lim, i As Integer
tn = InputBox("Enter table number:")
lim = InputBox("Enter table limit:")
For i = 1 To lim
    Print tn & "*" & i & "=" & tn * i
Next
End Sub
```

### Practical 17:

Write a program that finds the sum of squares of the integers from 1 to n where n is positive integer value entered by the user (i.e.  $\text{Sum} = 1^2 + 2^2 + 3^2 + \dots + n^2$ )

```
Private Sub Command1_Click()
```

```
Dim n, sum, i As Integer
```

```
n = InputBox("Enter a positive number.")
```

```
sum = 0
```

```
For i = 1 To n
```

```
    sum = sum + (i * i)
```

```
Next
```

```
Print sum
```

```
End Sub
```

### Practical 18:

Write a program to display  
counting from 10 to 1.

```
Private Sub Command1_Click()
```

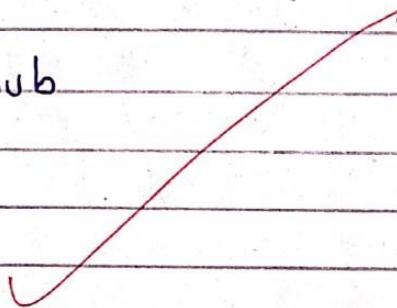
```
Dim i As Integer
```

```
For i = 10 To 1 Step -1
```

```
    Print i
```

```
Next
```

```
End Sub
```



Output:-

10  
9  
8  
7  
6  
5  
4  
3  
2  
1

Practical 19:

Write a program to print  
the following series:

1 4 7 10 .... 40

Private Sub Command1\_Click()

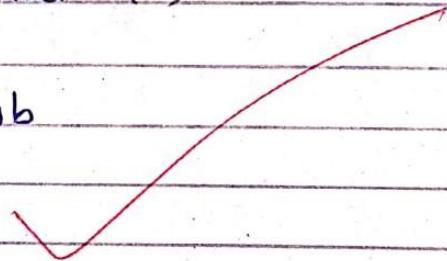
Dim i As Integer

For i = 1 To 40 Step 3

Print i;

Next

End Sub



Output:-

1 4 7 10 13 16 19 22 25 28 31 34 37 40

?

## Practical 20:

Write a program that inputs a number and displays whether it is a perfect square or not.

```
Private Sub Command1_Click()
```

```
    Dim number, squareRoot As Integer
```

```
    Dim squareRoot As Double
```

```
    number = InputBox("Enter a number.")
```

```
    squareRoot = Sqr(number)
```

```
    If Int(squareRoot) = squareRoot Then
```

```
        MsgBox number & " is a perfect sq."
```

```
    Else
```

```
        MsgBox number & " is not a perfect sq."
```

```
    End If
```

```
End Sub
```



## Practical 21:

Write a program to get an integer from the user and display whether it is prime or not.

```
Private Sub Command1_Click()
Dim i, n As Integer
n = InputBox("Enter a number")
For i = 2 To n - 1
    If n Mod i = 0 Then
        MsgBox("Number is not prime.")
        Exit For
    End If
Next
```

If i = n Then

MsgBox("Number is prime.")

End If

End Sub

## Practical 22:

Write a program that inputs two numbers and display their greatest common divisor (GCD) and least Common Multiple (LCM).

Private Sub Command1\_Click()

Dim n1, n2, gcd, lcm, i As Integer

n1 = InputBox("Enter First Number:")

n2 = InputBox("Enter Second Number.")

For i = 1 To n1

If n1 Mod i = 0 And n2 Mod i = 0 Then

gcd = i

End If

Next

lcm = (n1 \* n2) / gcd

MsgBox "The GCD is " & gcd & " AND LCM  
is " & lcm

End Sub

9/17/23

## Practical 23:

Write a program that gets five numbers from the user and store them in array. It then displays the numbers in reverse order.

```
Private Sub Command1_Click()
Dim arr(4) As Integer
Dim i As Integer
For i = 0 To 4
    arr(i) = InputBox("Enter a number:")
Next
Print "Original Array:"
For i = 0 To 4
    Print arr(i)
Next
Print "Array in reverse order:"
For i = 4 To 0 Step -1
    Print arr(i)
Next
End Sub
```

## Practical 24:

Write a program that stores five values in an integer array. It gets a number from the user and finds it in the array using sequential search.

```
Private Sub Command1_Click()
Dim arr(4) As Integer
Dim i, n, flag As Integer
flag = -1
For i = 0 To 4
    arr(i) = InputBox("Enter a number:")
```

Next

```
n = InputBox("Enter a number to search:")
For i = 0 To 4
    If n = arr(i) Then
        flag = i
        Exit For
    End If
```

Next

```
If flag = -1
```

MsgBox "Number not found", vbInformation

Else

MsgBox "Num found at " & flag,

End If

End Sub

vbInformation

## Practical 2-S:

Write a program that stores five values in an array. It sorts the array using bubble sort. It also displays the value of unsorted and sorted array.

```
Private Sub Command1_Click()
Dim arr(4) As Integer
Dim i, j, temp As Integer
For i = 0 To 4
    arr(i) = Val(InputBox("Enter a number:"))
Next
```

```
Print "Original Array"
```

```
For i = 0 To 4
```

```
    Print arr(i)
```

```
Next
```

```
For i = 0 To 4
```

```
    For j = 0 To 3
```

```
        If arr(i) > arr(j+1) Then
```

```
            temp = arr(j)
```

```
            arr(j) = arr(j+1)
```

```
            arr(j+1) = temp
```

```
    End If
```

```
Next
```

```
Next
```

```
Print "Sorted Array:"
```

```
For i = 0 To 4
```

```
    Print arr(i)
```

```
Next
```

```
End Sub
```

### Practical 26 q):

Write a program that gets ten numbers from the users in array and displays the maximum number.

```
Private Sub Command1_Click()
Dim arr(9) As Integer
Dim i, max As Integer
For i = 0 To 9
    arr(i) = Val(InputBox("Enter a Number"))
Next
max = arr(0)
For i = 0 To 9
    If arr(i) > max Then
        max = arr(i)
    End If
Next
MsgBox "Maximum Number is " & max
End Sub
```

## Practical 26 b):

Write a program that gets ten number from the user in array and displays the min number.

Private Sub Command1\_Click()

Dim arr(9) As Integer

Dim i As Integer

For i = 0 To 9

arr(i) = Val(InputBox("Enter Number:"))

Next

min = arr(0)

For i = 0 To 9

If arr(i) < min Then

min = arr(i)

End If

Next

MsgBox "Minimum Number is " & min

End Sub

## Practical 27:

Write a program that gets five numbers from the user in array and displays total number of even and odd numbers stored in the array.

```
Private Sub Command1_Click()
Dim arr(4) As Integer
Dim i, even, odd As Integer
For i = 0 To 4
    arr(i) = Val(InputBox("Enter number:"))
Next
even = 0
odd = 0
For i = 0 To 4
    If arr(i) Mod 2 = 0 Then
        even = even + 1
    Else
        odd = odd + 1
    End If
Next
MsgBox "Total even: " & even & vbCrLf
    "Total odd: " & odd
End Sub
```

## Practical 28:

Write a program that stores five names in an array of strings. It gets a name from the user and displays whether the name is present in the array or not.

```
Private Sub Command1_Click()
    Dim arr(4) As String
    Dim n As String
    Dim i As Integer
    For i = 0 To 4
        arr(i) = InputBox("Enter a name:")
    Next
```

```
n = InputBox("Enter name to find:")
For i = 0 To 4
    If arr(i) = n Then
        MsgBox n & " found at index " & i
        Exit Sub
    End If
```

```
Next
MsgBox n & " not found"
End Sub
```

✓  
25/02/23

### Practical #29:

Write a program that gets the name of the user and displays its length.

```
Private Sub Command1_Click()
Dim n As String
n = InputBox("Enter your name:")
MsgBox Len(n)
End Sub
```

### Practical #30:

Write a program that inputs your date of birth and displays your age.

```
Private Sub Command1_Click()
```

```
D1 = "04/09/2003"
```

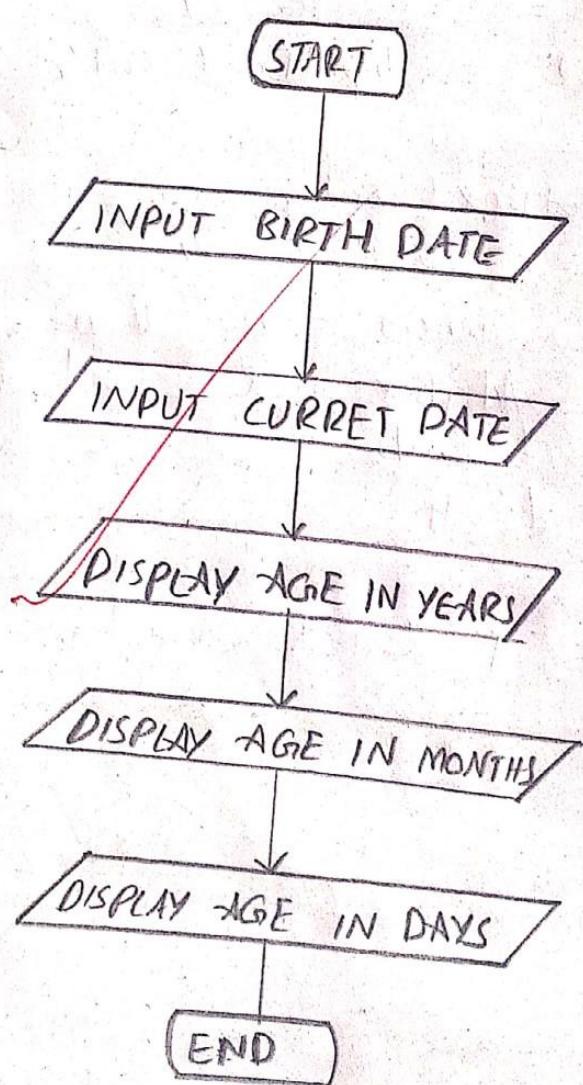
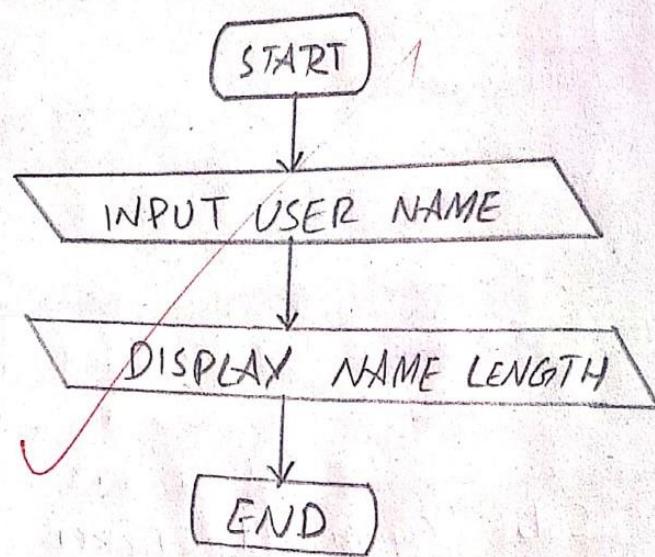
```
D2 = "06/16/2023"
```

```
Print "Age in Years:"; DateDiff("yyyy", D1, D2)
```

```
Print "Age in Months:"; DateDiff("m", D1, D2)
```

```
Print "Age in Days:"; DateDiff("d", D1, D2)
```

```
End Sub
```



Practical # 31 :-

Write a VB6 program to  
Draw the following pattern

1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5

```
Private Sub Command1_Click()
    Dim i, j As Integer
    For i = 1 To 5
        For j = 1 To 5
            Print j;
        Next j
        Print
    Next i
End Sub
```

Output:-

1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5  
1 2 3 4 5

**Practical # 32:**  
Write a VB6 program to Draw  
the following Pattern:

1 5 5 5 5 5  
2 2 4 4 4 4  
3 3 3 3 3 3  
4 4 4 4 2 2  
5 5 5 5 5 1

```
Private Sub Command1_Click()
Dim i, j, k As Integer
For i = 1 To 5
    For j = 1 To i
        print i;
    Next j
    For k = 6 - i To 1 Step -1
        print 6 - i;
    Next k
    print
Next
End Sub
```

Output:-

1 5 5 5 5 5  
2 2 4 4 4 4  
3 3 3 3 3 3  
4 4 4 4 2 2  
5 5 5 5 5 1

### Practical # 53:

Input two matrices of order  
3x3 and ADD them in a  
separate matrix

```
Private Sub Command1_Click()
Dim i, j, A(2,2), B(2,2), C(2,2) As Integer
For i = 0 To 2
    For j = 0 To 2
        A(i,j) = Val(InputBox("Enter values in A(" & i & "," & j & ")"))
    Next j
Next i
For i = 0 To 2
    For j = 0 To 2
        B(i,j) = Val(InputBox("Enter value in B(" & i & "," & j & ")"))
    Next j
Next i
For i = 0 To 2
    For j = 0 To 2
        C(i,j) = A(i,j) + B(i,j)
    Next j
Next i
Print "Matrix A", "Matrix B", "Matrix C"
For i = 0 To 2
    For j = 0 To 2
        Print A(i,j), B(i,j), C(i,j)
    Next j
Next i
End Sub
```

### Practical # 34:

Write a VB6 program that accepts 10 integer and store them in an array. The program determine whether the certain elements are duplicate/repet or not.

```
Private Sub Command1_Click()
    Dim i, j, A(9) As Integer
    For i = 0 To 9
        A(i) = Val(InputBox("Enter value in A(" & i & ")"))
    Next i
    For i = 0 To 9
        For j = i + 1 To 9
            If A(i) = A(j) Then
                MsgBox "Duplicate value found!"
                Exit Sub
            End If
        Next j
    Next i
    MsgBox "No Duplicate Value found!"
End Sub
```

### Practical # 35:

Write a VB6 program that checks whether the entered number is perfect or Not.

```
Private Sub Command1_Click()
```

```
Dim n As Integer
```

```
n = Val(InputBox("Enter a Number:"))
```

```
Call Perfect(n)
```

```
End Sub
```

```
Sub Perfect(n)
```

```
Dim p, i As Integer
```

```
For i = 1 To n - 1
```

```
If n Mod i = 0 Then
```

```
p = p + 1
```

```
End If
```

Next

```
If p = n Then
```

```
MsgBox "Number is perfect!"
```

```
Else
```

```
MsgBox "Number is not Perfect!"
```

```
End If
```

```
End Sub
```

## Practical # 36:

### a) Draw a Hollow Diamond

```
Private Sub Command1_Click()
```

```
Dim i, j, s As Integer
```

```
For i = 1 To 9 Step 2
```

```
    For s = 1 To 9 - i Step 2
```

```
        Print " ";
```

```
    Next s
```

```
    For j = 1 To i
```

```
        If j ≠ 1 Or j = i Then
```

```
            Print "*";
```

```
        Else
```

```
            Print " ";
```

```
        End If
```

```
    Next j
```

```
    Print
```

```
Next i
```

```
For i = 7 To 1 Step -2
```

```
    For s = (7 - i) To 1 Step -2
```

```
        Print " ";
```

```
    Next s
```

```
    For j = i To 1 Step -1
```

```
        If j = 1 Or j = i Then
```

```
            Print "*";
```

```
        Else
```

```
            Print " ";
```

```
    End If
```

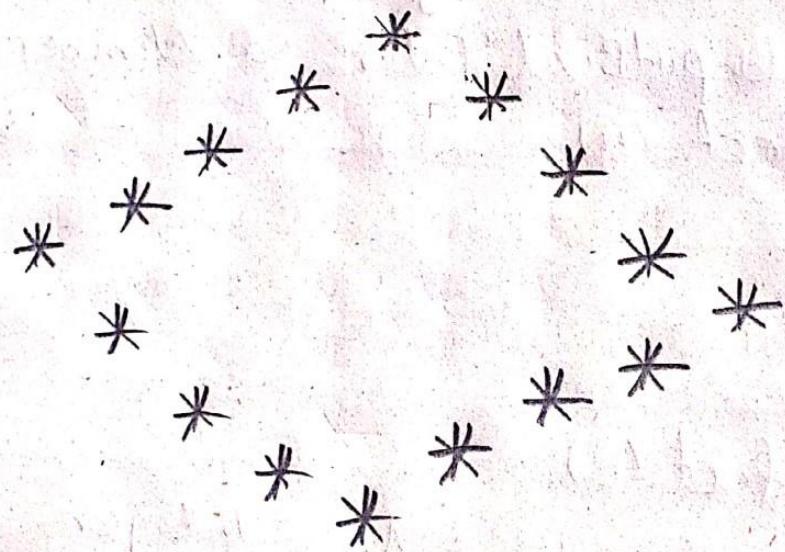
```
    Next j
```

```
    Print
```

```
Next i
```

```
End Sub
```

Output:-



## Practical #36

### b) Draw a full Diamond

```
Private Sub Command1_Click()
```

```
Dim i, j, s As Integer
```

```
For i = 1 To 9 Step 2
```

```
    For j = 1 To 9-i Step 2
```

```
        Print " ";
```

```
    Next s
```

```
    For j = 1 To i
```

```
        Print "*";
```

```
    Next j
```

```
    Print
```

```
    Next i
```

```
    Print " ";
```

```
    For i = 7 To 1 Step -2
```

```
        For s = 7-i To 1 Step -2
```

```
            Print " ";
```

```
        Next s
```

```
        For j = i To 1 Step -1
```

```
            Print "*";
```

```
        Next j
```

```
        Print
```

```
    Next i
```

```
End Sub
```

Output:-

\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*  
\* \* \* \* \*