CHIANG MAI UNIVERSITY
College of Arts, Media and Technology
1st Semester / Academic Year 2025
960101 Fundamentals of Programming Logic in Digital Industry

Lab Assignment 09: More Nested Loop

Name ....กันต์ธีร์ วารีสอาด................... Student ID .... Section......1......

Objectives:

1) The student understands the concept of Nested Loop
2) The student can appropriately choose and apply the control structure.

Note that the nested structure **must be used** .

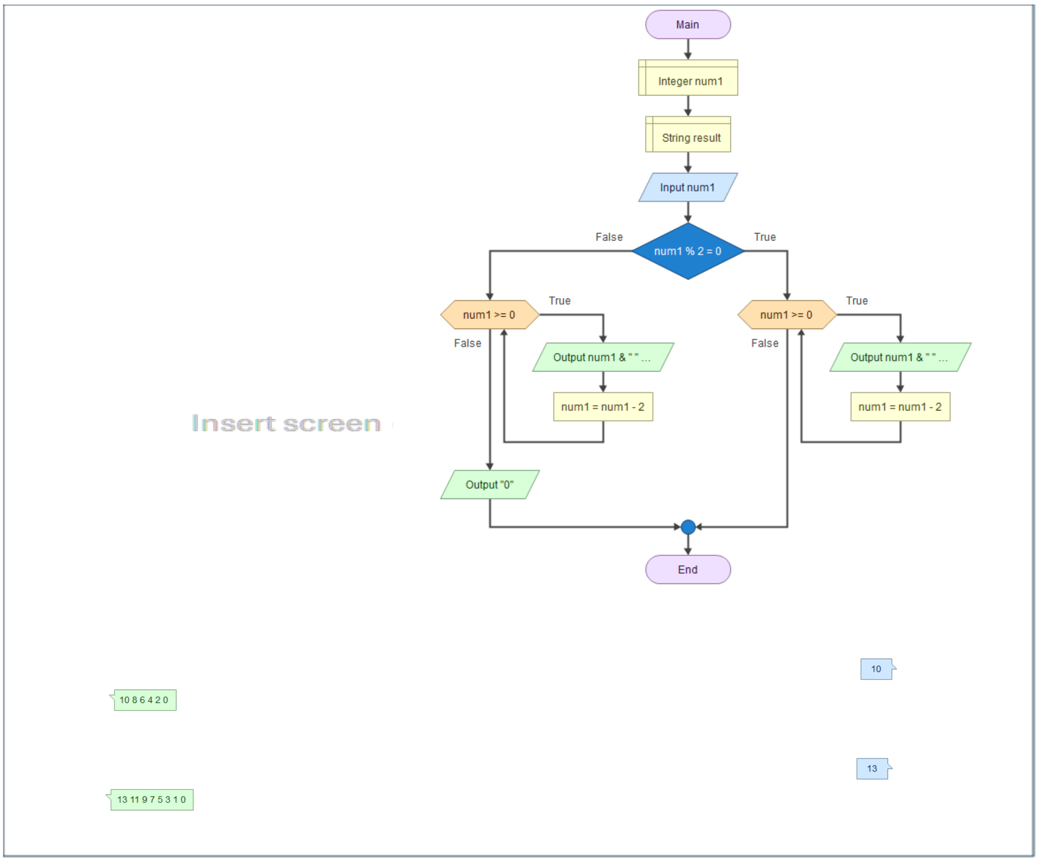
Problems sets

1. Develop a program to accept an integer from the user and output a series of numbers decreasing by 2 until the value reaches 0.
   For example,

   If the user inputs **10**, the program will display **10 8 6 4 2 0**

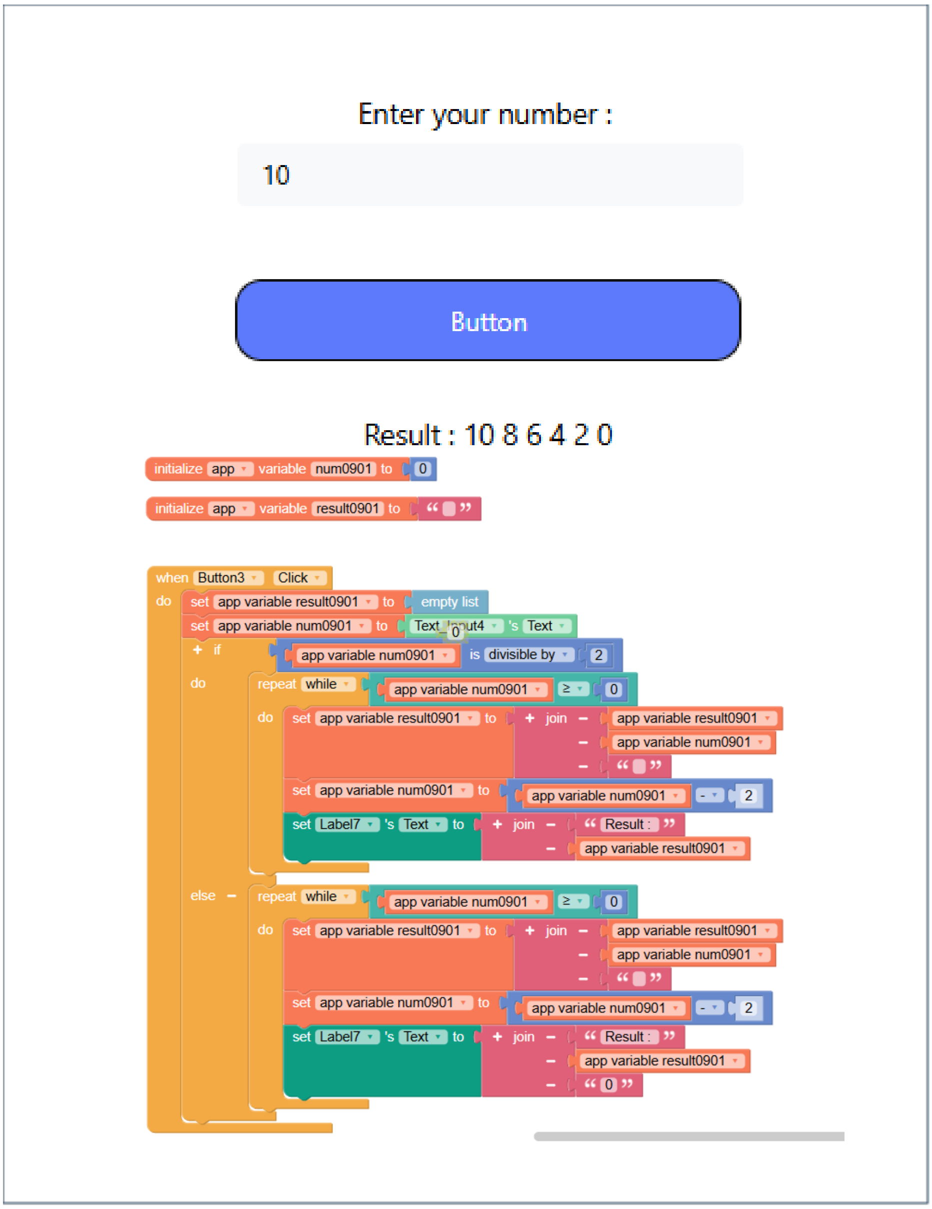
   If the user inputs **13**, the program will display **13 11 9 7 5 3 1 0**

   1.1 Draw a flowchart of the program using **Flowgorithm**.



```
10
10 8 6 4 2 0
13
13 11 9 7 5 3 1 0
```

1.2 Develop the program in **Thunkable** (Always create new screen!!).

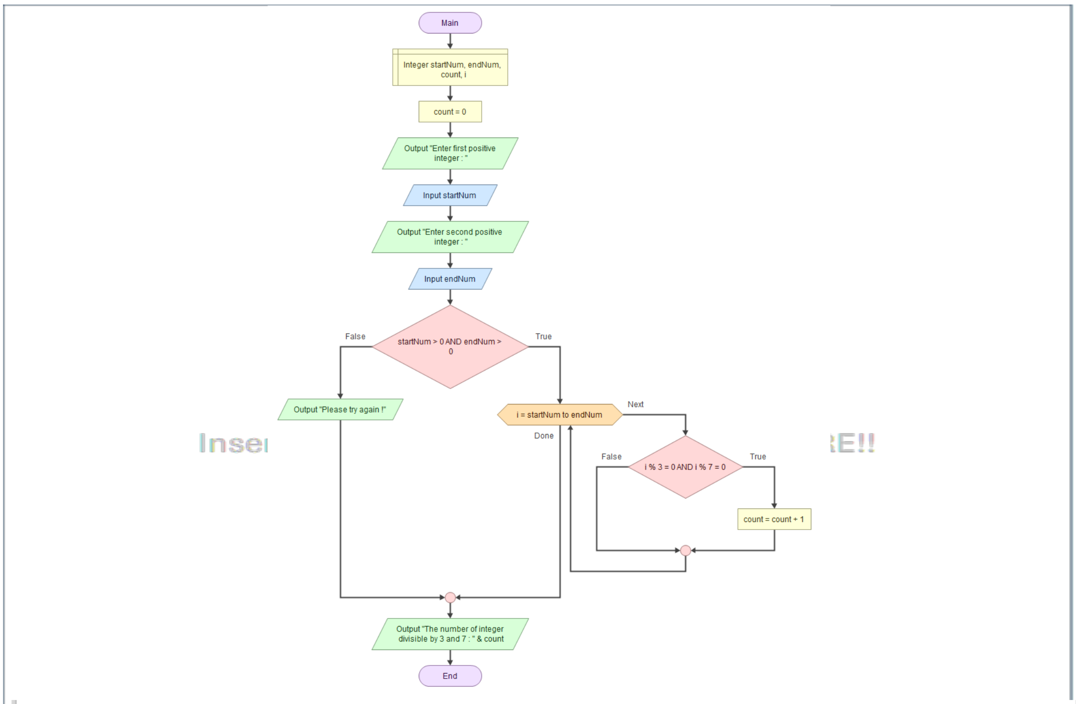


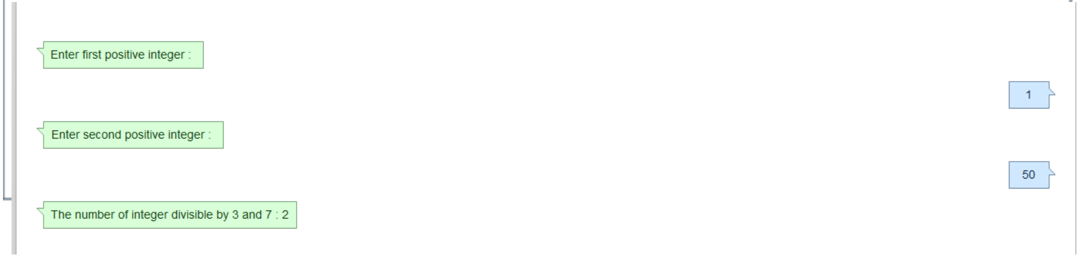
Result : 10 8 6 4 2 0

```
initialize app variable num0901 to 0

initialize app variable result0901 to " "

when Button3 Click
do  set app variable result0901 to empty list
    set app variable num0901 to Text_Input4 's Text
    if  app variable num0901 is divisible by 2
    do  repeat while app variable num0901 ≥ 0
        do  set app variable result0901 to join  app variable result0901
                                                 app variable num0901
                                                 " "
            set app variable num0901 to app variable num0901 - 2
            set Label7 's Text to join  "Result : "
                                        app variable result0901
    else
        repeat while app variable num0901 ≥ 0
        do  set app variable result0901 to join  app variable result0901
                                                 app variable num0901
                                                 " "
            set app variable num0901 to app variable num0901 - 2
            set Label7 's Text to join  "Result : "
                                        app variable result0901
                                        "0"
```

2. Create a program to receive 2 positive integer numbers from user and count the number between the input numbers that is divisible by 3 and 7.
   For example,
   If the user inputs **1** and **50**, the program will display **2.** (21 and 42 are divisible by 3 and 7)

   2.1 Create a flowchart on **Flowgorithm.**
   Answer:

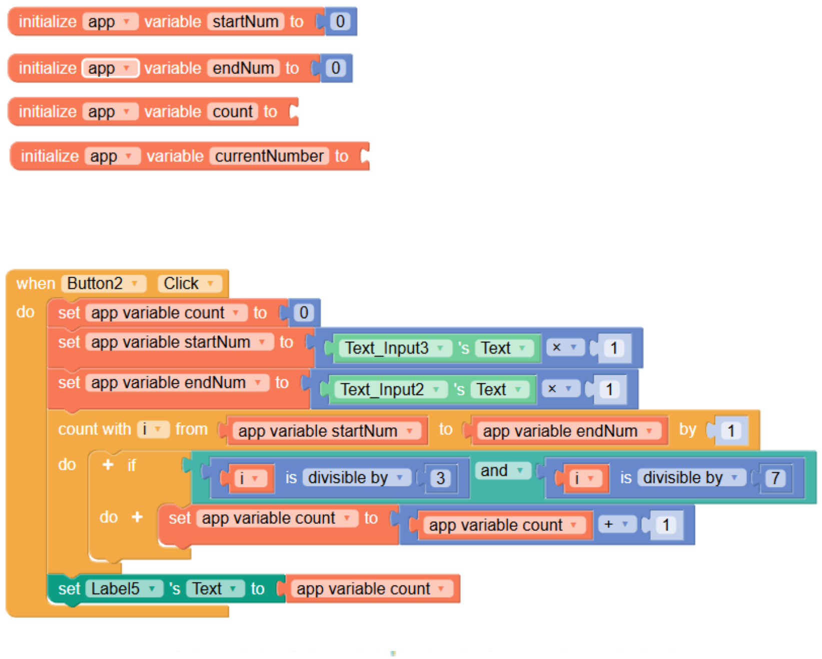

Main
Integer startNum, endNum, count, i
count = 0
Output "Enter first positive integer : "
Input startNum
Output "Enter second positive integer : "
Input endNum
False
startNum > 0 AND endNum > 0
True
Output "Please try again !"
i = startNum to endNum
Next
Done
False
i % 3 = 0 AND i % 7 = 0
True
count = count + 1
Output "The number of integer divisible by 3 and 7 : " & count
End



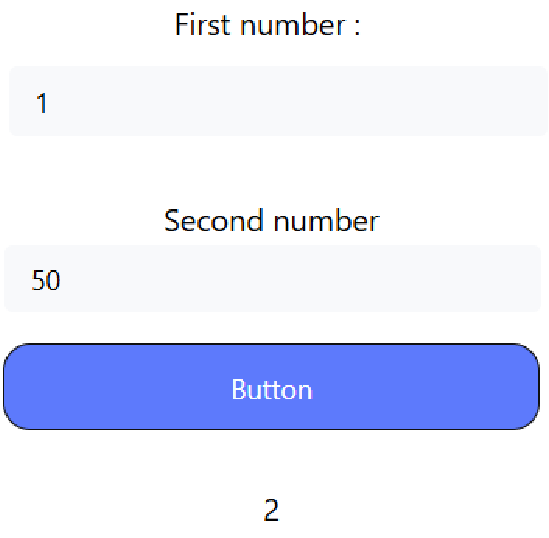
```
Enter first positive integer :
                                                        1
Enter second positive integer :
                                                        50
The number of integer divisible by 3 and 7 : 2
```

2.2 Create a program on **Thunkable**.

Answer:



```
initialize app variable startNum to 0
initialize app variable endNum to 0
initialize app variable count to
initialize app variable currentNumber to

when Button2 Click
do  set app variable count to 0
    set app variable startNum to Text_Input3 's Text × 1
    set app variable endNum to Text_Input2 's Text × 1
    count with i from app variable startNum to app variable endNum by 1
    do  + if  i is divisible by 3 and i is divisible by 7
          do + set app variable count to app variable count + 1
    set Label5 's Text to app variable count
```



First number :

1

Second number

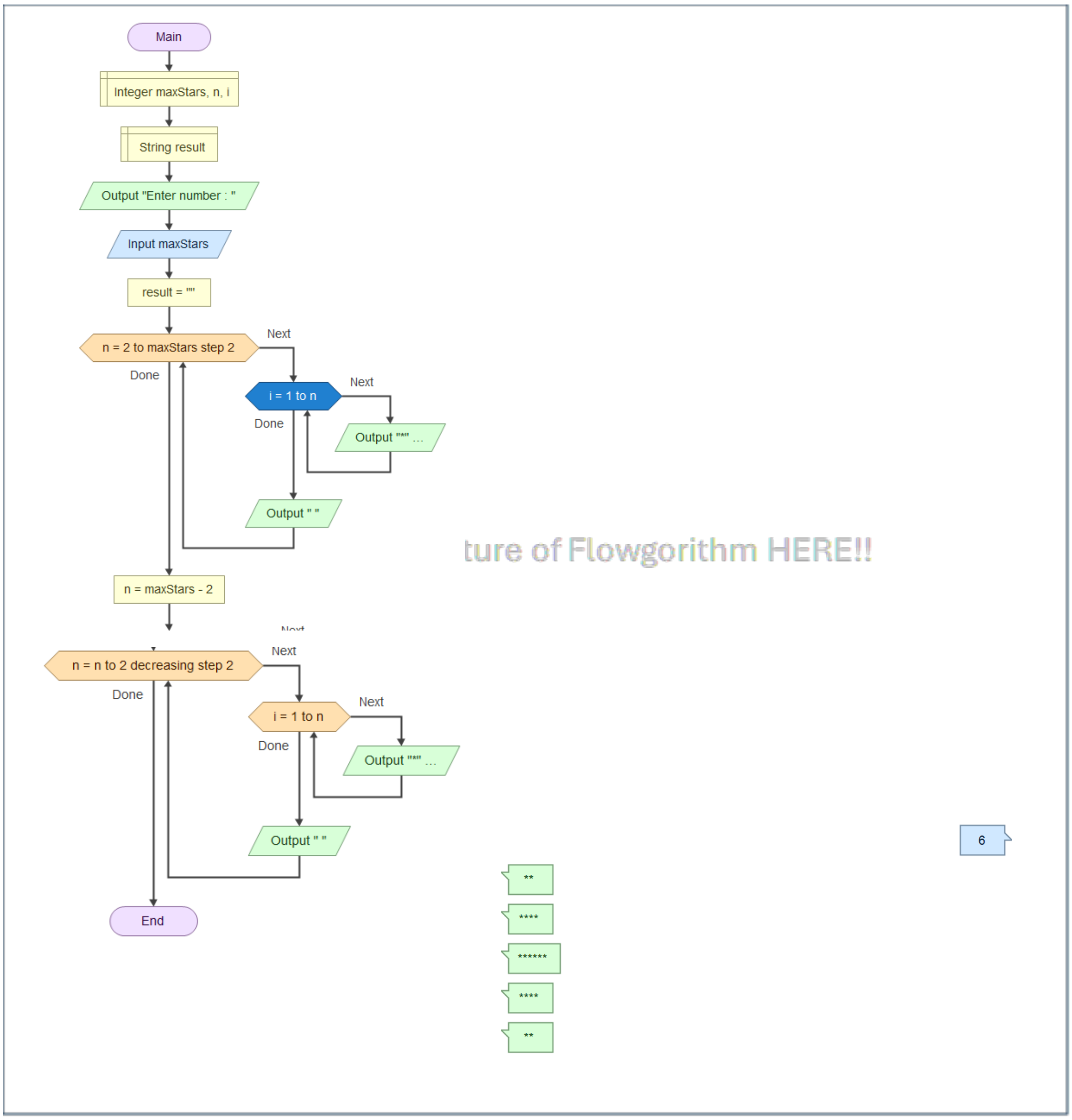
50

2

3. Create a program to display a triangle with lines of *. The user needs to input the number of maximum *. Then the output will display lines of * only the even number of *.
For example, if the user input 6, the program will display

```
**
****
******
****
**
```

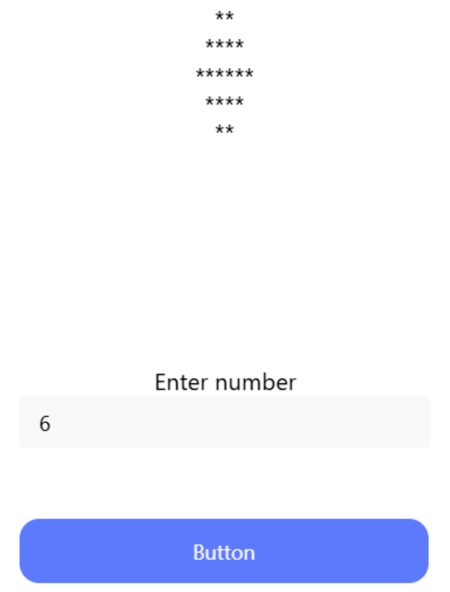
3.1 Create a flowchart on **Flowgorithm.**
Answer:



```
**
****
******
****
**
```

3.2 Create a program on **Thunkable**.

Answer:

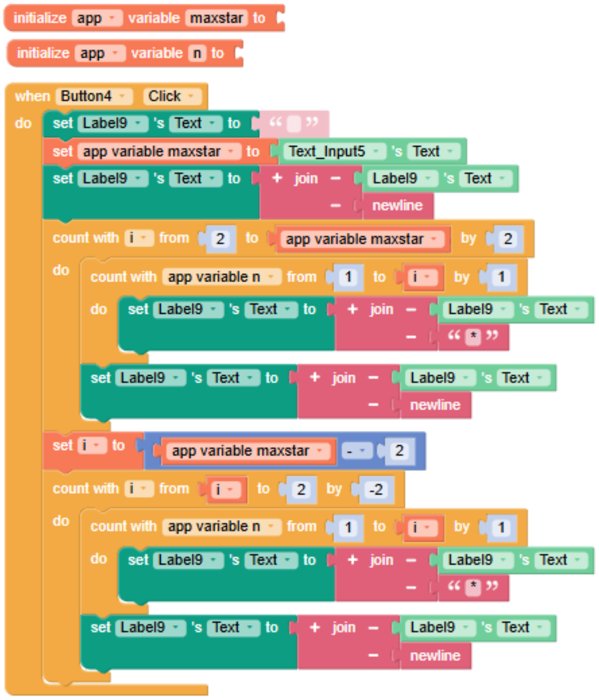


```
**
****
******
****
**
```

Enter number

6

Insert screen capture of Thunkable HERE!!



```
initialize app variable maxstar to
initialize app variable n to

when Button4 Click
do  set Label9 's Text to " "
    set app variable maxstar to Text_Input5 's Text
    set Label9 's Text to + join – Label9 's Text
                                 – newline
    count with i from 2 to app variable maxstar by 2
    do  count with app variable n from 1 to i by 1
        do  set Label9 's Text to + join – Label9 's Text
                                         – " * "
        set Label9 's Text to + join – Label9 's Text
                                     – newline
    set i to app variable maxstar - 2
    count with i from i to 2 by -2
    do  count with app variable n from 1 to i by 1
        do  set Label9 's Text to + join – Label9 's Text
                                         – " * "
        set Label9 's Text to + join – Label9 's Text
                                     – newline
```

4. Create a program to receive 2 positive integer numbers from user as N and M then display a multiplication table of N to M.

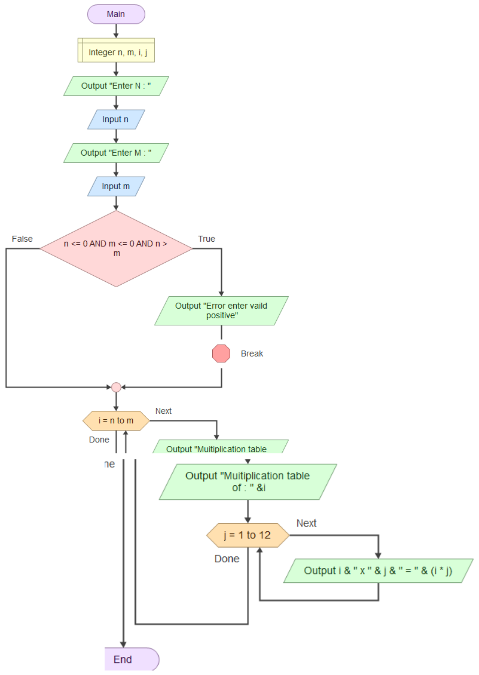
For example N = 2, M = 4

Result: 2 X 1 = 2, 2 X 2 = 4,....., 2 X 12 = 24
3 X 1 = 3, 3 X 2 = 6,....., 3 X 12 = 36
4 X 1 = 4, 4 X 2 = 8,....., 4 X 12 = 48

Remark: you can display multiplication table in any format

4.1 Create a flowchart on **Flowgorithm.**

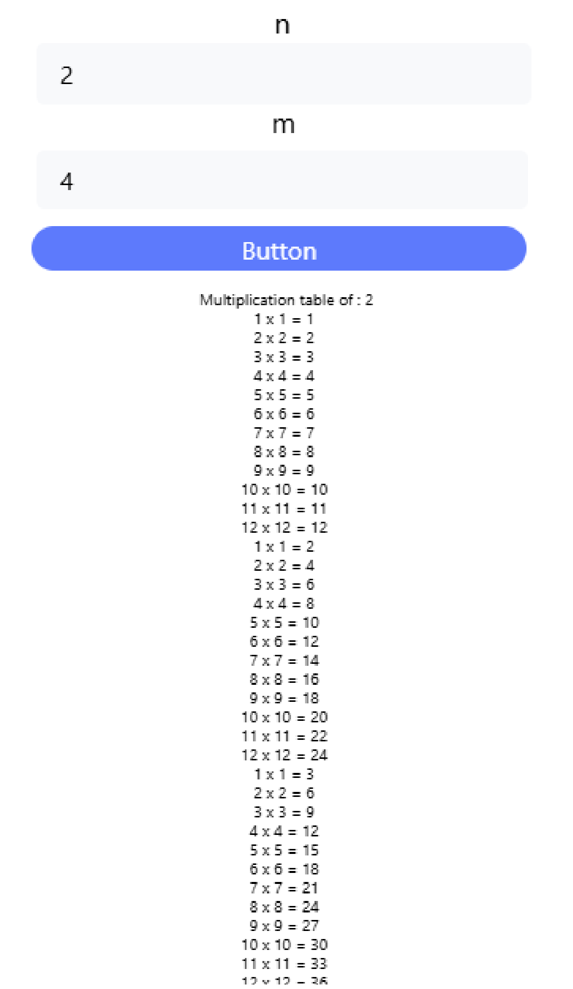
Answer:


Main
Integer n, m, i, j
Output "Enter N : "
Input n
Output "Enter M : "
Input m
False
n <= 0 AND m <= 0 AND n > m
True
Output "Error enter vaild positive"
Break
i = n to m
Next
Done
Output "Muitiplication table
ne
Output "Muitiplication table of : " &i
j = 1 to 12
Next
Done
Output i & " x " & j & " = " & (i * j)
End

f Flowgorithm HERE!!

4.2 Create a program on **Thunka**

Answer:



n

2

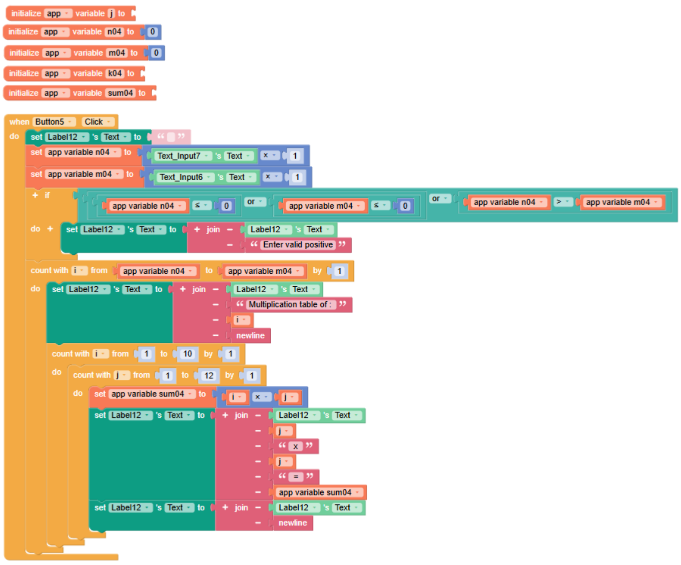
m

4

Button

```
Multiplication table of : 2
1 x 1 = 1
2 x 2 = 2
3 x 3 = 3
4 x 4 = 4
5 x 5 = 5
6 x 6 = 6
7 x 7 = 7
8 x 8 = 8
9 x 9 = 9
10 x 10 = 10
11 x 11 = 11
12 x 12 = 12
1 x 1 = 2
2 x 2 = 4
3 x 3 = 6
4 x 4 = 8
5 x 5 = 10
6 x 6 = 12
7 x 7 = 14
8 x 8 = 16
9 x 9 = 18
10 x 10 = 20
11 x 11 = 22
12 x 12 = 24
1 x 1 = 3
2 x 2 = 6
3 x 3 = 9
4 x 4 = 12
5 x 5 = 15
6 x 6 = 18
7 x 7 = 21
8 x 8 = 24
9 x 9 = 27
10 x 10 = 30
11 x 11 = 33
12 x 12 = 36
```

Insert scree



```
initialize app variable j to
initialize app variable n04 to 0
initialize app variable m04 to 0
initialize app variable k04 to
initialize app variable sum04 to

when Button5 Click
do  set Label12 's Text to " "
    set app variable n04 to Text_Input7 's Text × 1
    set app variable m04 to Text_Input6 's Text × 1
    if  app variable n04 ≤ 0 or app variable m04 ≤ 0 or app variable n04 > app variable m04
    do  set Label12 's Text to join Label12 's Text
                                    "Enter valid positive"
    count with i from app variable n04 to app variable m04 by 1
    do  set Label12 's Text to join Label12 's Text
                                    "Multiplication table of : "
                                    i
                                    newline
        count with i from 1 to 10 by 1
        do  count with j from 1 to 12 by 1
            do  set app variable sum04 to i × j
                set Label12 's Text to join Label12 's Text
                                            j
                                            "x"
                                            j
                                            "="
                                            app variable sum04
                set Label12 's Text to join Label12 's Text
                                            newline
```