**PYTHON**

**PROJECT REPORT**

**21. BITWISE OPERATIONS CALCULATOR**

**Submitted by**

|  |  |  |
| --- | --- | --- |
| Roll No. | Registration No. | Name |
| RK21PBA28 | 12113509 | Aniruddha Das |
| RK21PBA30 | 12103093 | Jalluri Uday Bhaskar |
| RK21PBB58 | 12113020 | Pula Karthik |

**Submitted to**

**Dr. Ramandeep Sandhu**

Section: K21PB

**DISCIPLINE OF CSE/IT**

**LOVELY SCHOOL OF COMPUTER SCIENCE & ENGG.**

**LOVELY PROFESSIONAL UNIVERSITY, PHAGWARA**



**DECLARATION**

We, **Aniruddha Das, Jalluri Uday Bhaskar and Pula Karthik**, students of B.tech (CSE) under CSE/IT Discipline at, Lovely Professional University, Punjab, hereby declare that all the information furnished in this project report is based on our own intensive work and is genuine.

Date: 14 - 11 - 2022 Signature:

Registration Nos. 12113509 – 12103093 - 12113020

**TABLE OF CONTENTS**

1. Introduction
2. Objectives
3. Design
4. Algorithms Used
5. Result Screenshot
6. Conclusion
7. References

**1. INTRODUCTION**

This mini project is a GUI implementation of a bitwise (AND, OR, XOR & NOT) calculator using Tkinter with Python. Tkinter was written by Steen Lumholt and Guido van Rossum, then later revised by Fredrik Lundh. Unlike in command line interface, everything in Graphical User Interface is meant to be user friendly, software products nowadays almost always come with a GUI interface such that users are able to better interact with the system.

Tkinter: The Tkinter package/library is the standard python A.P.I through which we can effortlessly create GUI applications, it is de facto GUI interface that comes pre-packaged with Python, generally, although one can easily install it using pip module otherwise, through a command line interface

Tkinter has various widgets (GUI elements) which a GUI needs to function, they are:

* Buttons
* Labels
* Radio button
* Combo box
* Entry
* Frame
* Notebook

…and many others, these GUI elements or widgets can be placed in the Tkinter main event loop, note that every application we use, present in out computers, is essentially an event loop that loops on and on, infinitely until we press the required ‘stop’ button, which in many cases is denoted by a X(cross). This is how a GUI based application is made, though design part of it takes a lot of thought and effort to make it seamless to the user, here we have made a GUI application to showcase bitwise operations through Tkinter’s helpful widgets.

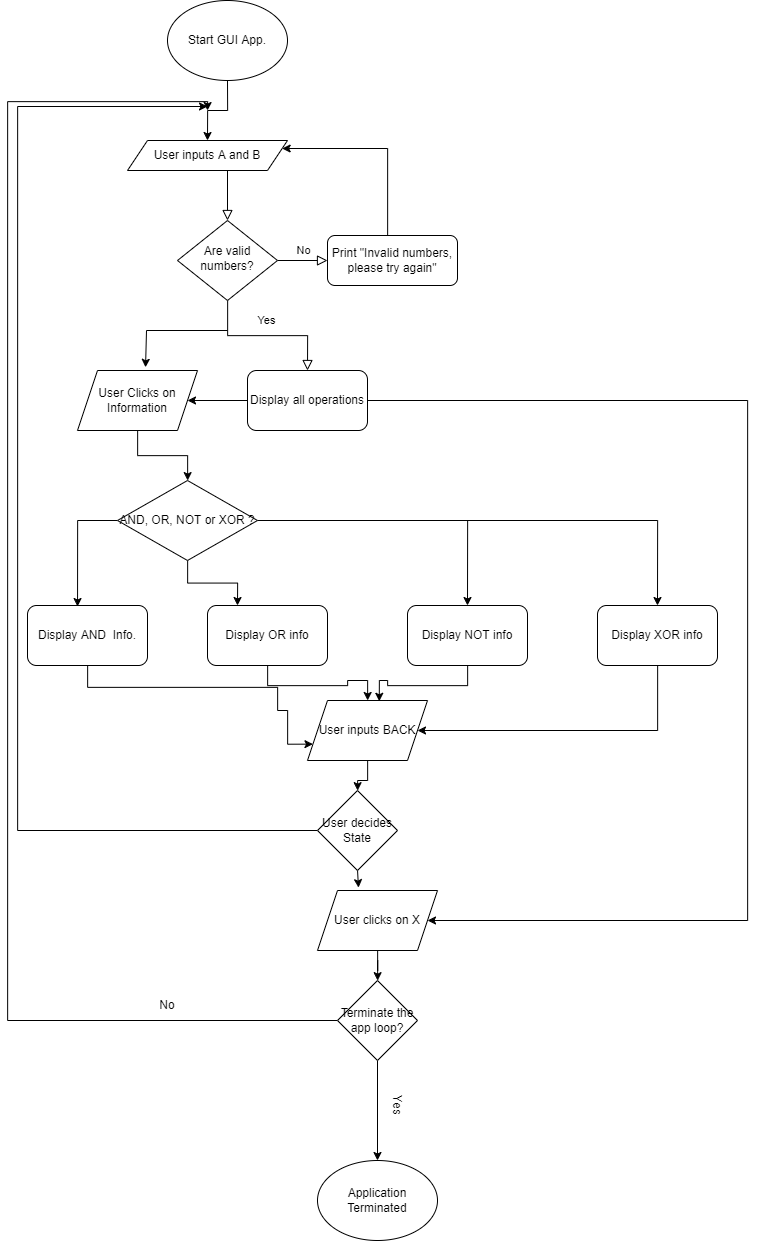
**2. OBJECTIVE**

Certain objectives we kept in mind while developing the application are given below:

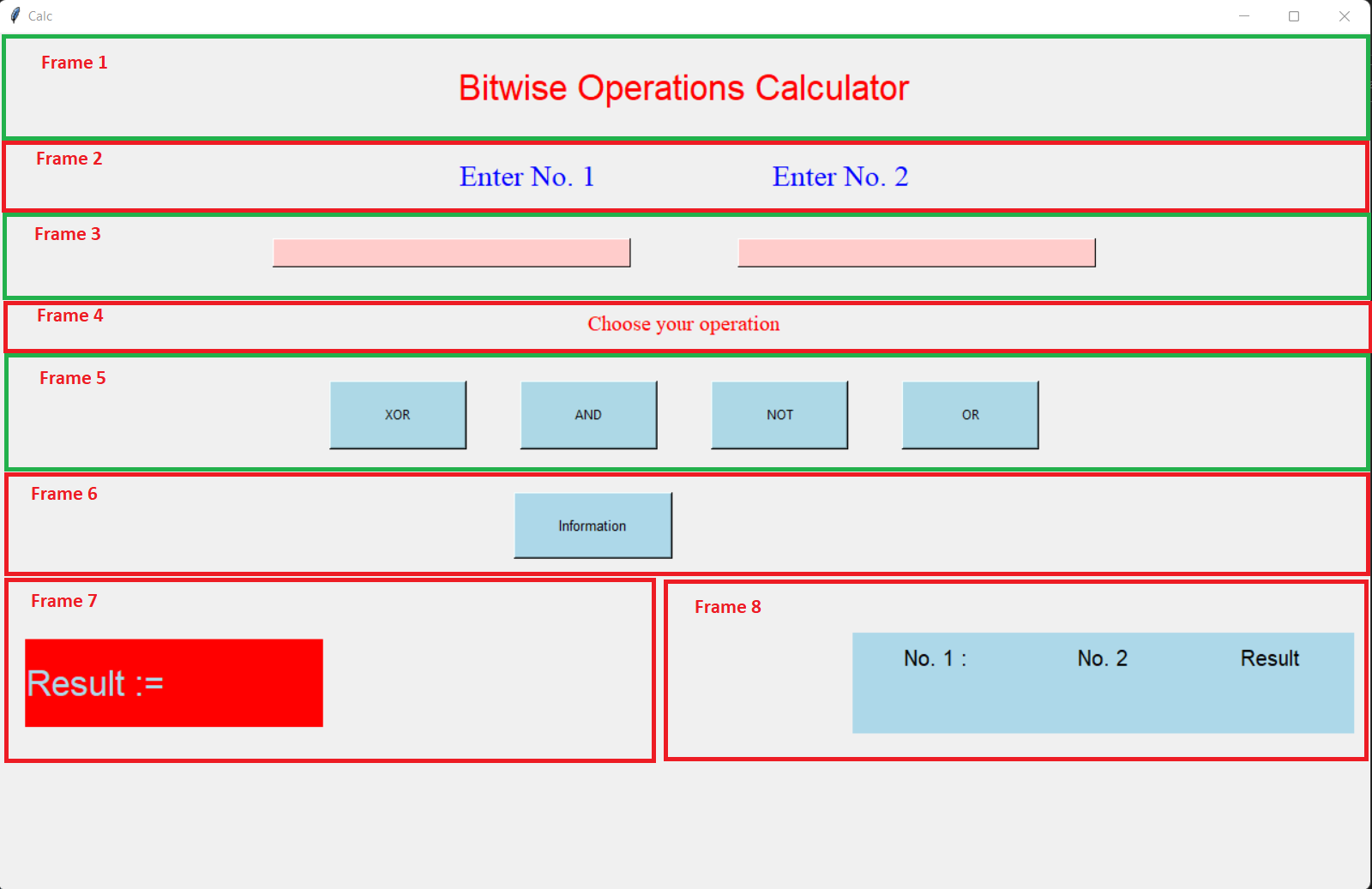
* Make every text element easy to read with an easily-visible font and specific colour along with it.
* Widgets should be clearly distinguished from each other using various colours and styles of widgets (raised, flat, dotted etc).
* Follow Gannt Chart and maintain a clear dead line.
* Check every piece of changed software for bugs, glitches etc.
* Co-ordinate and update main file base among group members.
* Implement various widgets as per requirement.
* Implement functions to convert numbers to binary.
* Implement functions to get XOR, AND, OR & NOT result and display it in the interface.

**3. DESIGN**

The design of gui application can be understood easily by the below given flowchart:



We divided each element of the GUI into various frames which can be packed to form a coherent application as elaborated by the following image:



…And then we used buttons and labels in each frame to structure the application correctly

Similarly, when a user presses information button, it will anchor to the corresponding ‘More Information’ window through the main event loop. The frame elements also use grid method to place text and string elements in it, uniformly.

Also, StringVar subclass from Tkinter is used to set or display some variable in labels and buttons as well, this is how we display results of bitwise operations between two numbers given number.

Now, there are Four bitwise operations: XOR, AND, OR and NOT, and as the name implies, 4 operators are used to represent them in python, they are ^, &, | and ~ respectively, operation happens at a bitwise level and is based on the following tables:

* XOR operation

|  |  |  |
| --- | --- | --- |
| A | B | Result |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

* AND operation

|  |  |  |
| --- | --- | --- |
| A | B | Result |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

* OR operation

|  |  |  |
| --- | --- | --- |
| A | B | Result |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

* NOT operation

|  |  |  |
| --- | --- | --- |
| A | B | Result |
| 0 | 0 | (0, 0) |
| 0 | 1 | (0, -1) |
| 1 | 0 | (-1, 0) |
| 1 | 1 | (-1, -1) |

Functions and modules for the same operations are made duly according to the need of the application.

**4. Algorithms used**

Only one algorithm was used, that was converting a number into binary

The algorithm is given below,

CONVBINARY (set number a):

Set temp = a

Set chr, a string

If a = 0:

Return string 0000

While temp > 0:

Set chr = chr + string of (temp % 2)

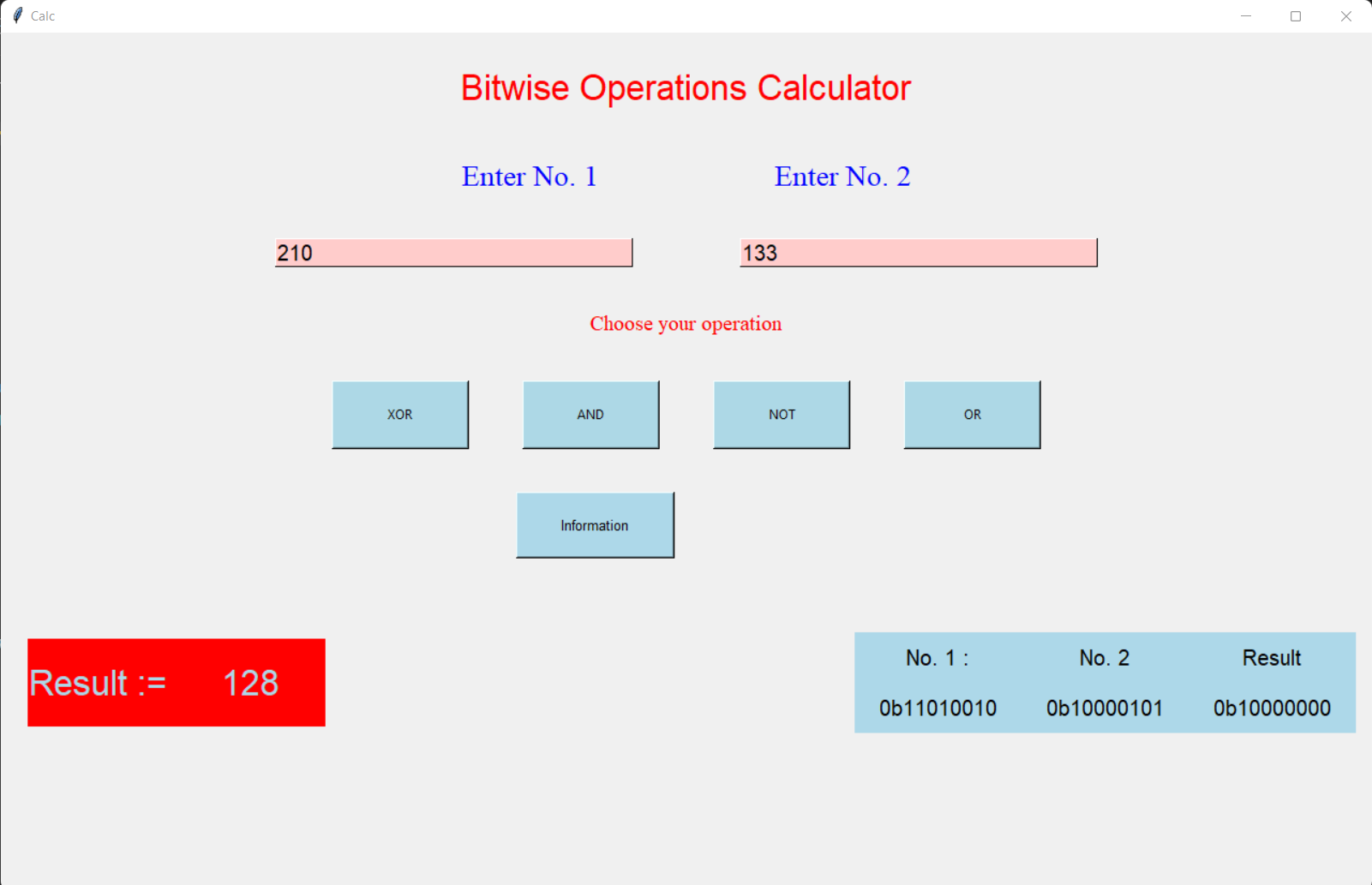
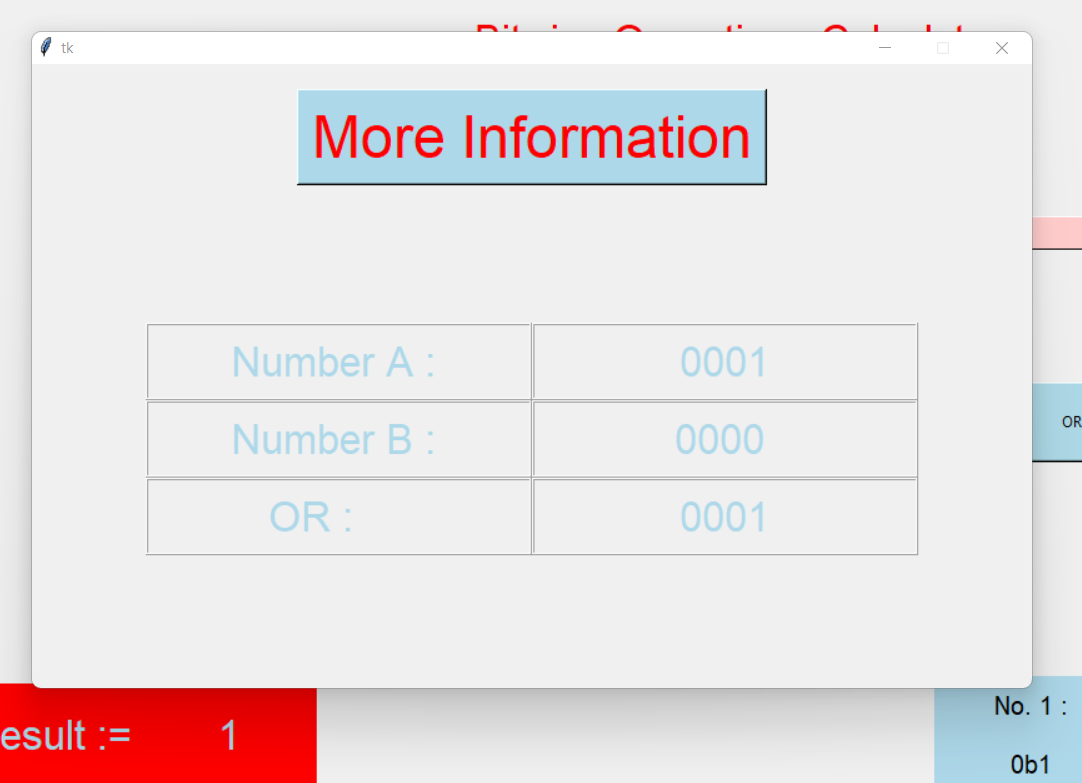
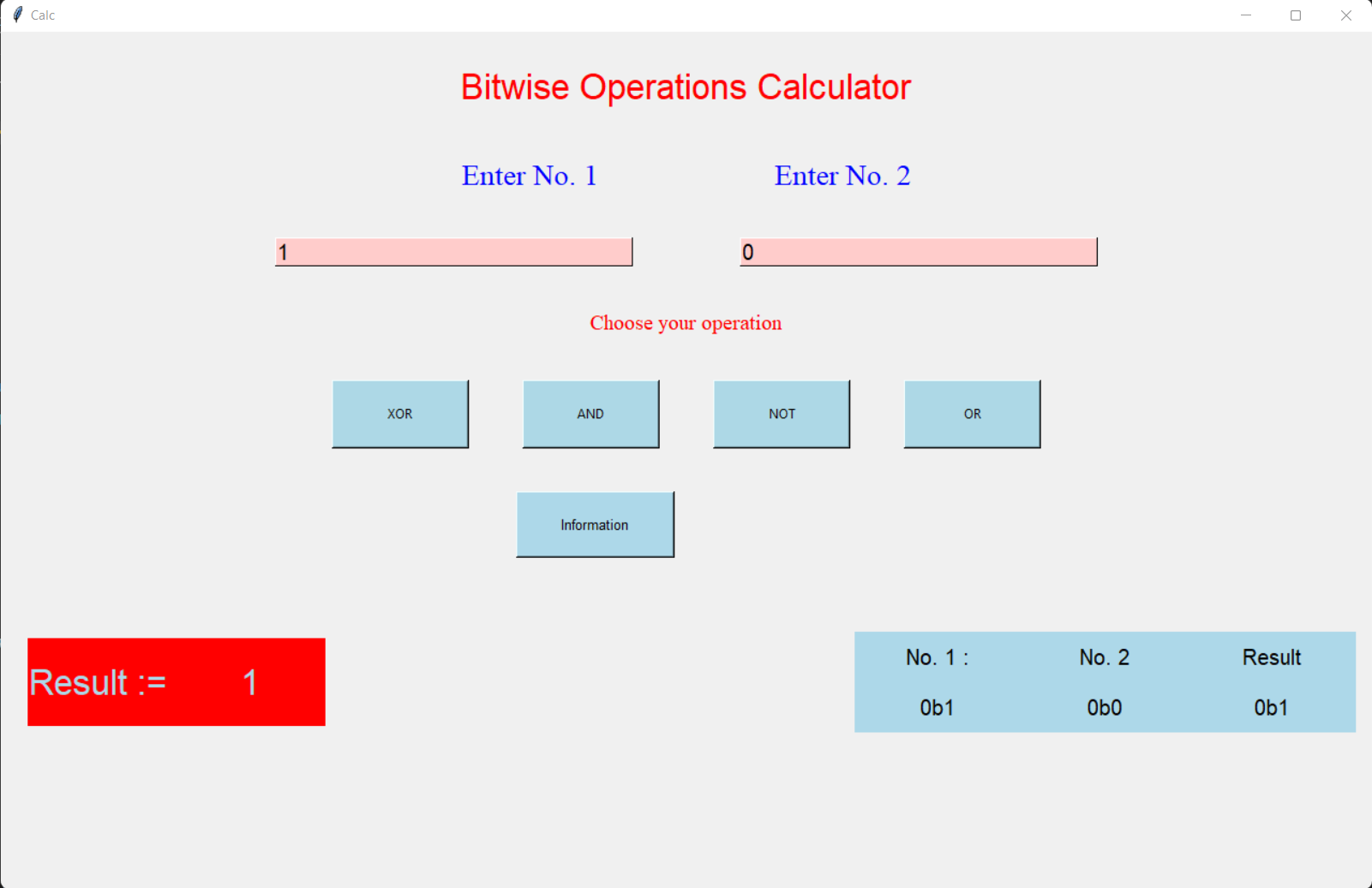
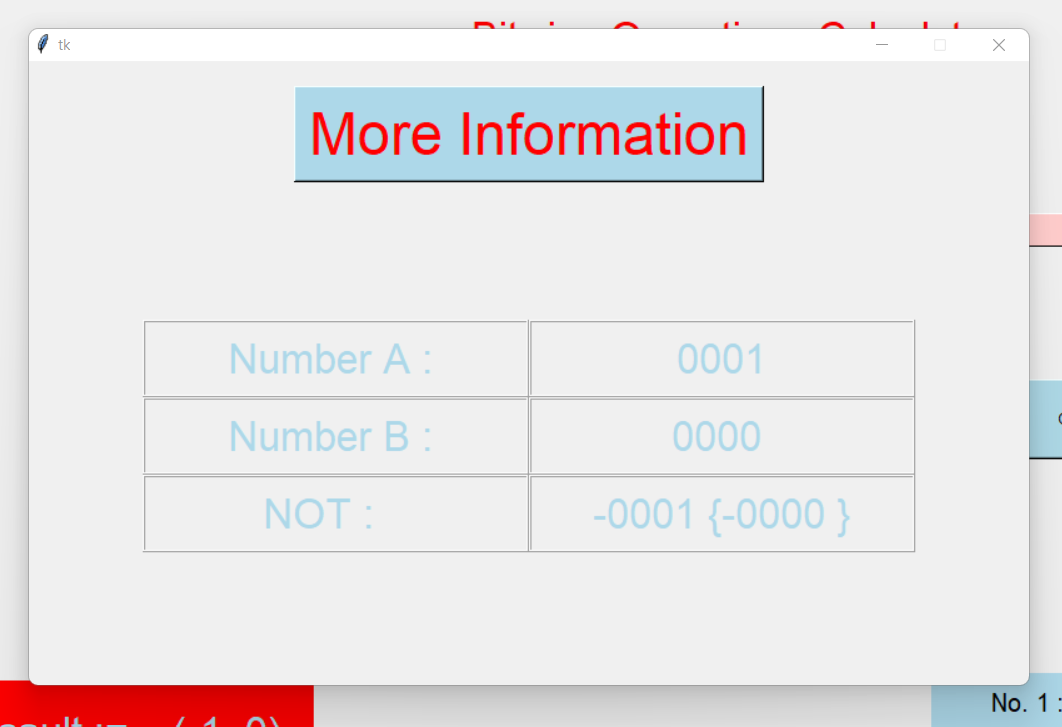
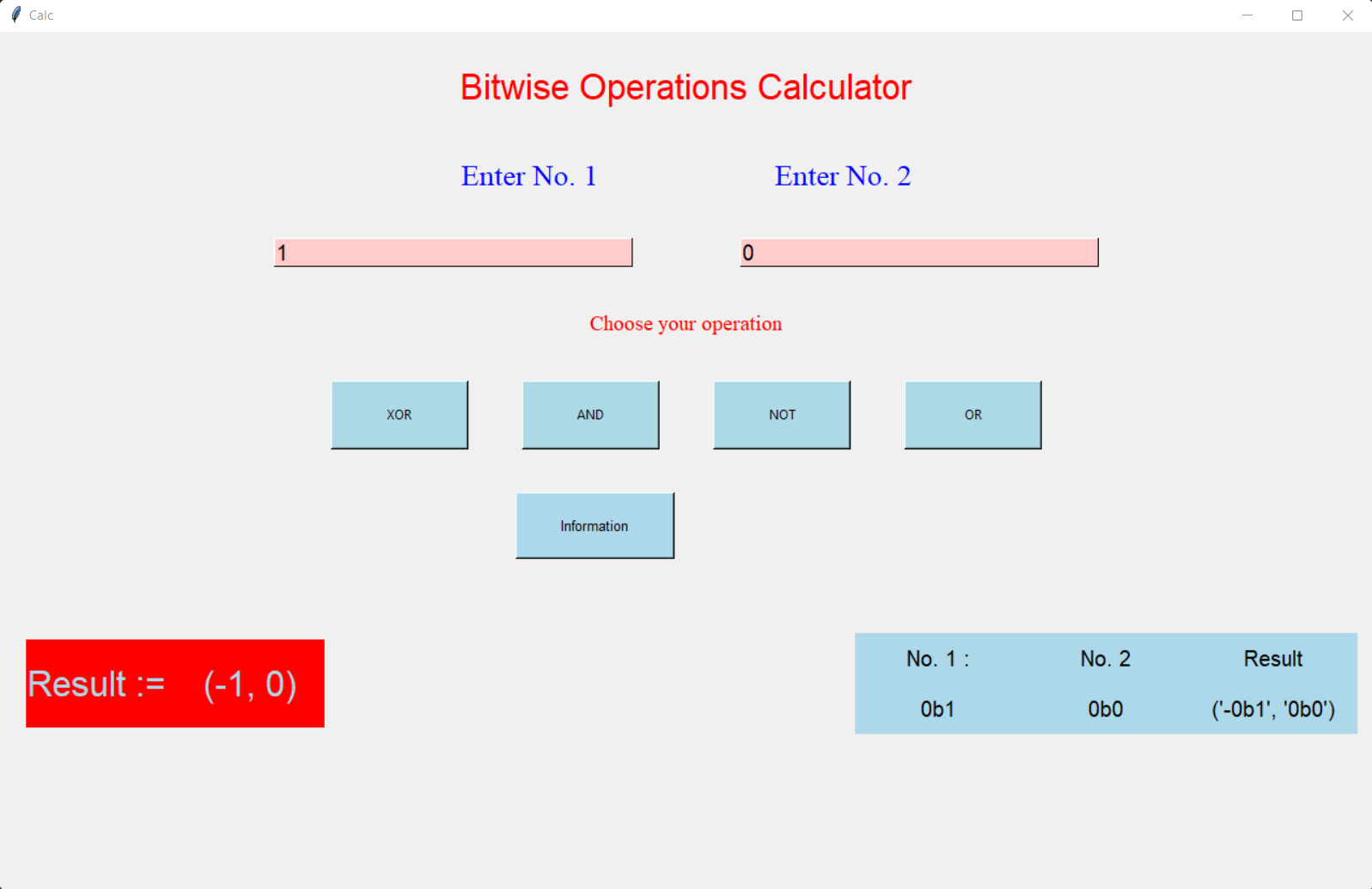
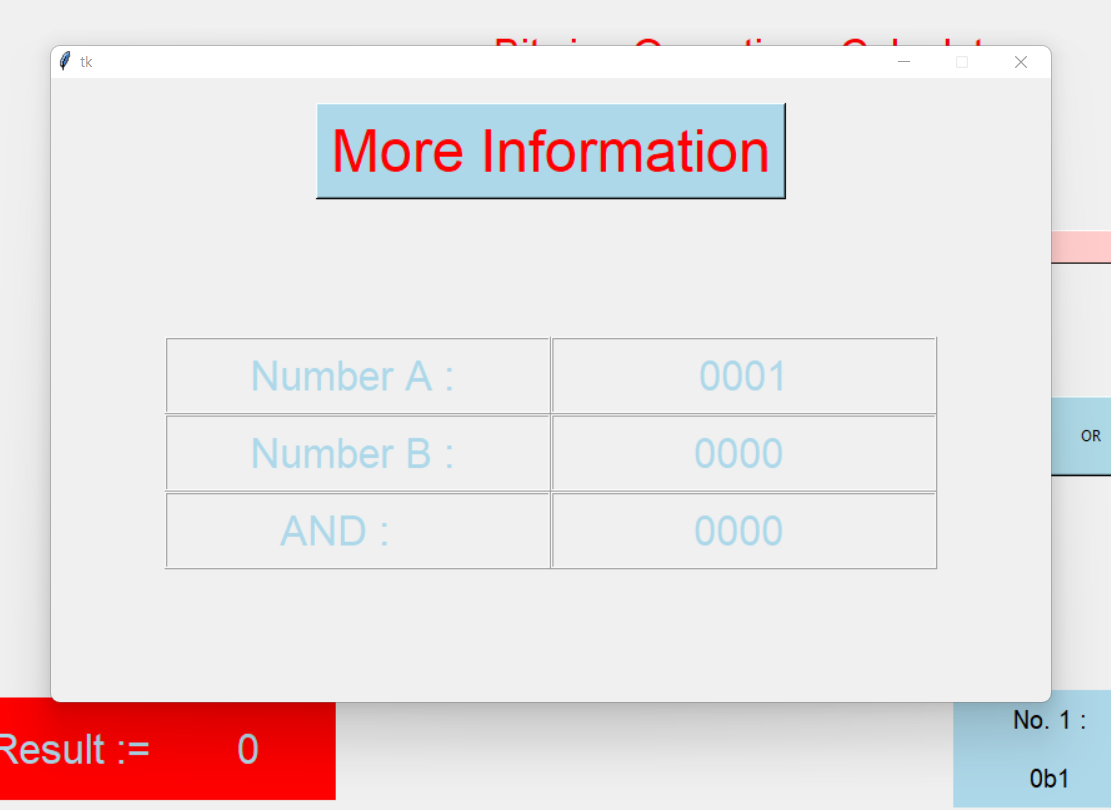
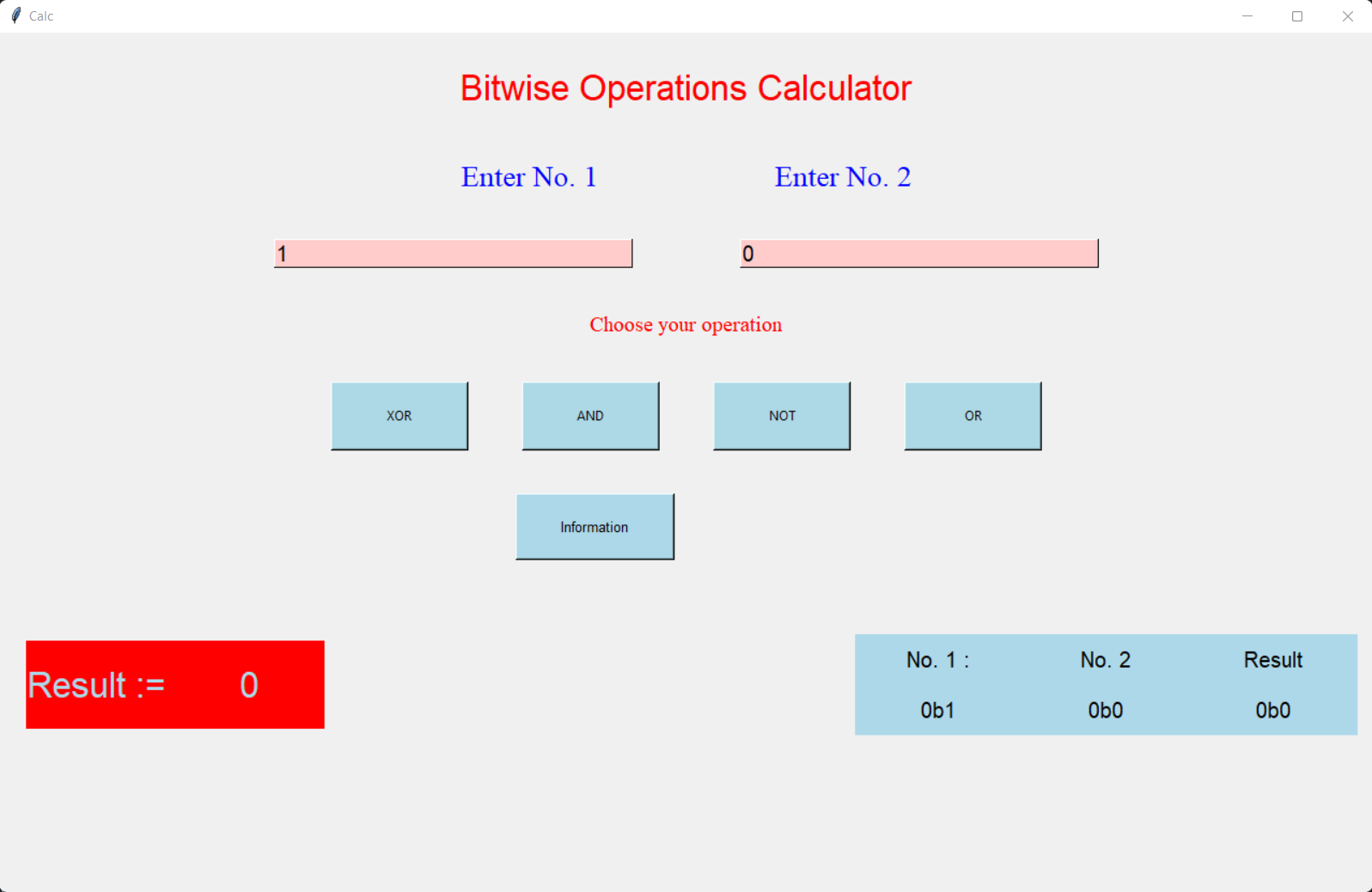
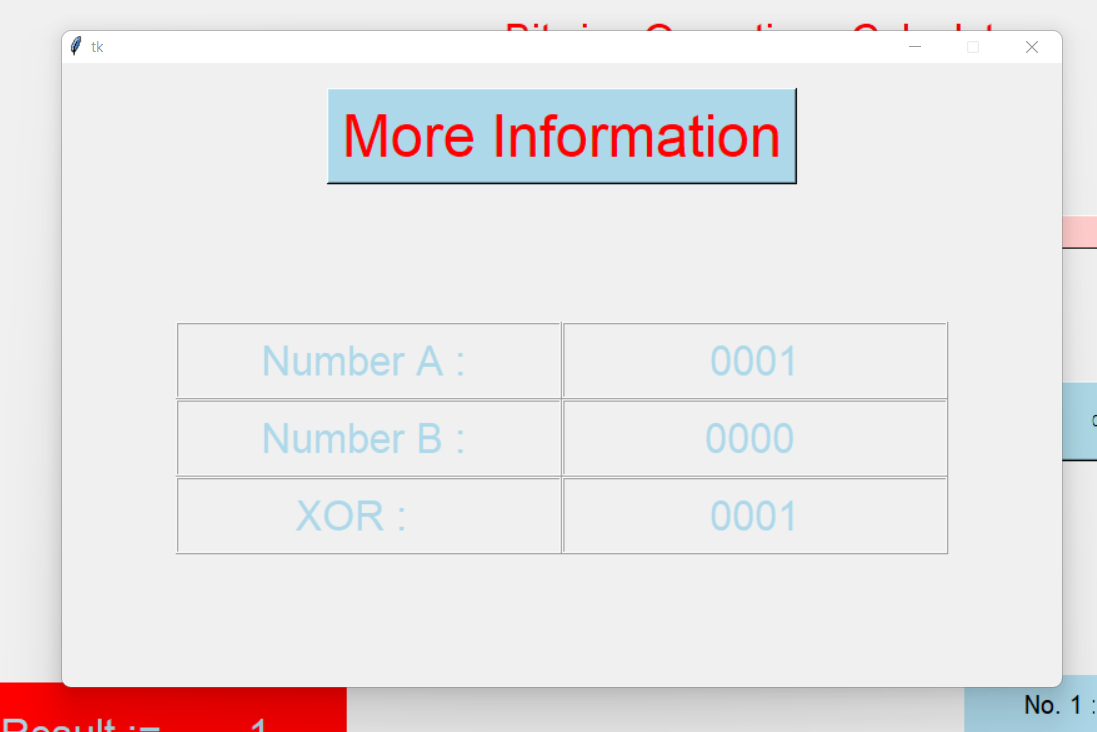
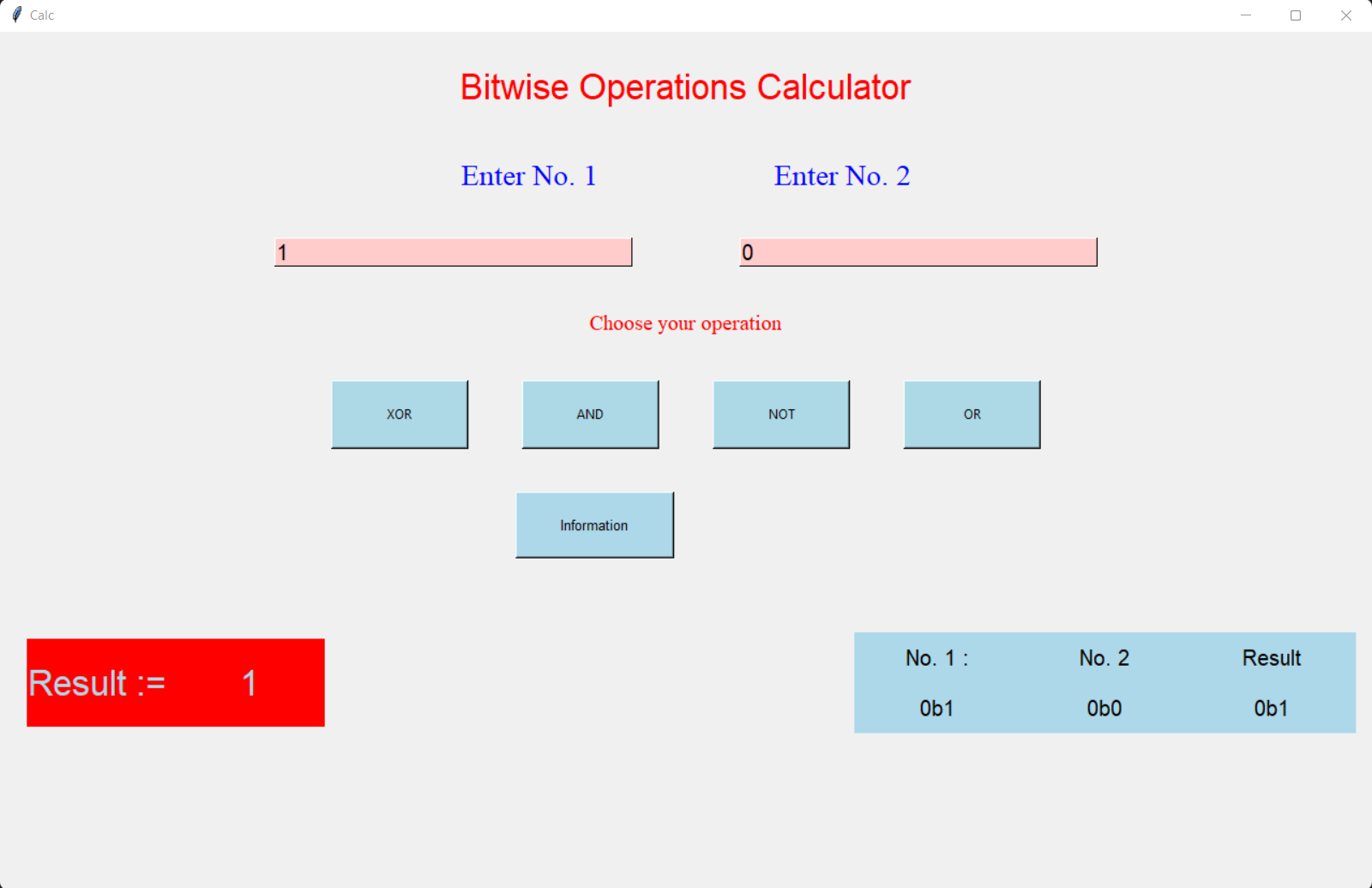
Set temp = temp // 2

If length(chr) < 2:

Set chr = chr + string of (0\*(4 – length of chr))

Return inverse of chr.

**5. RESULT SCREENSHOT**

****

**6. Conclusion**

This project taught us a lot about practical application of widgets and labels to form a coherent GUI, so that we can practically use it in real world. We also learnt about Framing and packing of data to make the GUI elements clean and easy to read, to pack data correctly and hold it in place we can use pack, grid and place method, moreover learnt how to get and ‘set’ data in label during main event loop. All in all, this project was a great learning experience for all of us that has helped us practice various aspects of Tkinter.

**7. References**

* <https://docs.python.org/3/library/tkinter.html>
* <https://en.wikipedia.org/wiki/Tkinter>
* <https://www.tutorialspoint.com/python/python_gui_programming.htm>
* <https://www.geeksforgeeks.org/python-gui-tkinter/>
* <https://pythonbasics.org/tkinter/>

…