EECE455 – Project #3

By Anthony Saab, Peter Farah and Karim Ghaddar

Official Description from Dr. Ali El Hussein

Project-3. ***Polynomial Arithmetic*** **(More Challenging; More Grades)**

The code will allow a user to perform arithmetic on polynomials in GF(2^m) with m = 163. The operations to be performed are modulo reduction, finding the inverse, addition, subtraction, multiplication, and division.

The implementation should be modular and can be easily extended to other standard powers such as: 163, 233, 239, 283, 409, 571.

The software should allow inputs and display outputs in either Binary or Hexadecimal.

Introduction:

This app is a GF(2^m) calculator currently supporting 2 <= m <= 514 and m=571 (m is an Integer). It can add, subtract, multiply, divide, inverse and reduce (apply modulo of irreducible polynomial). It can take Binary input or Hex input and display the result of the operation in decimal, binary and hex.

In the submitted zip, we will have the code in an archive file, and an executable.

Please make sure to disable any anti-virus, so that the executable can run properly.

Please view our demo video where we detail how to run the program in 2 methods: https://youtu.be/k2099KMwW8A

GUI:

Push buttons on the left side of the app, each button specifies what it does (add, subtract, multiply, divide, inverse, reduce)

Input boxes on the right with labels to their left specifying what they do. You can input either binary or hex in their respective input boxes. The app will give a warning when you input hex and binary so will only operate on either 2 hex or 2 bin but not one hex one bin, or for improper input format or improper input combinations (add takes 2 inputs; inverse takes one input; cannot divide by 0; binary1 improper input…)

The results are displayed in 3 formats: Decimal, Binary, and Hexadecimal.

The irreducible polynomial used is also displayed.

Backend:

We used: https://pypi.org/project/galois

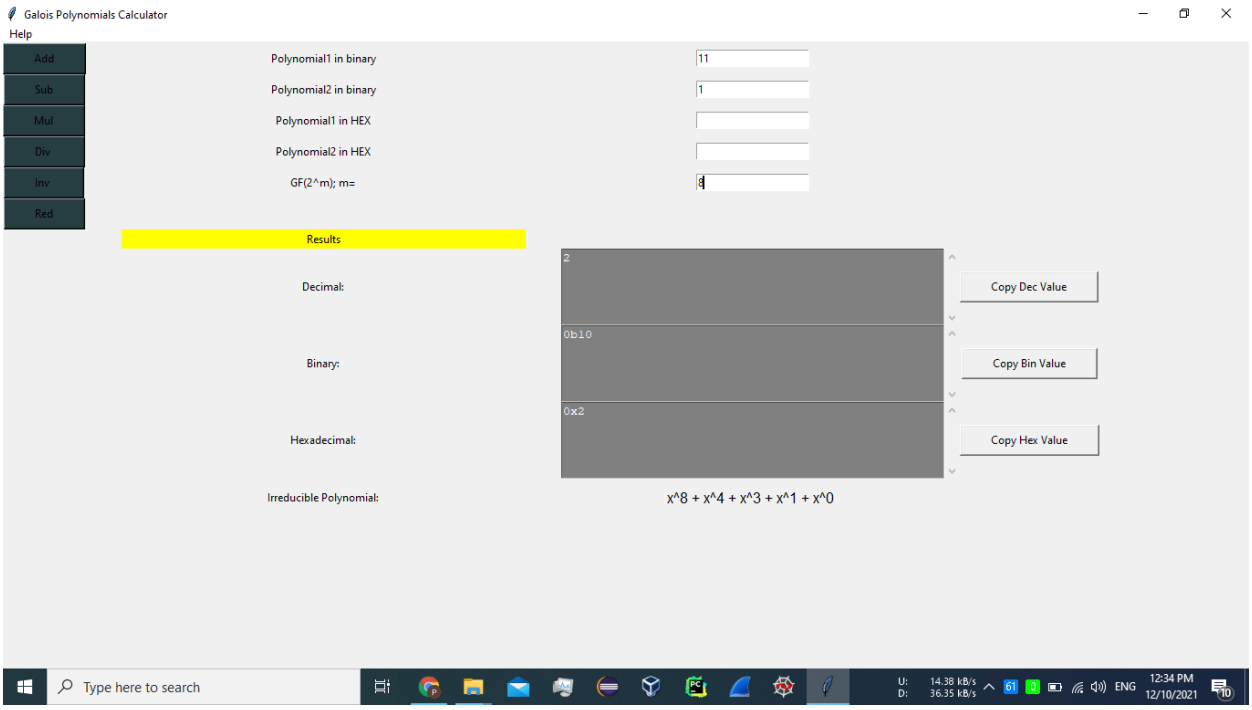
We used the previously mentioned Galois module to find all irreducible polynomials up to GF(2^514) and store them in a JSON file.

When doing an operation, the app now is pretty much instantaneous since it does not have to calculate the irreducible polynomial, it just looks it up in the JSON file which is very quick.

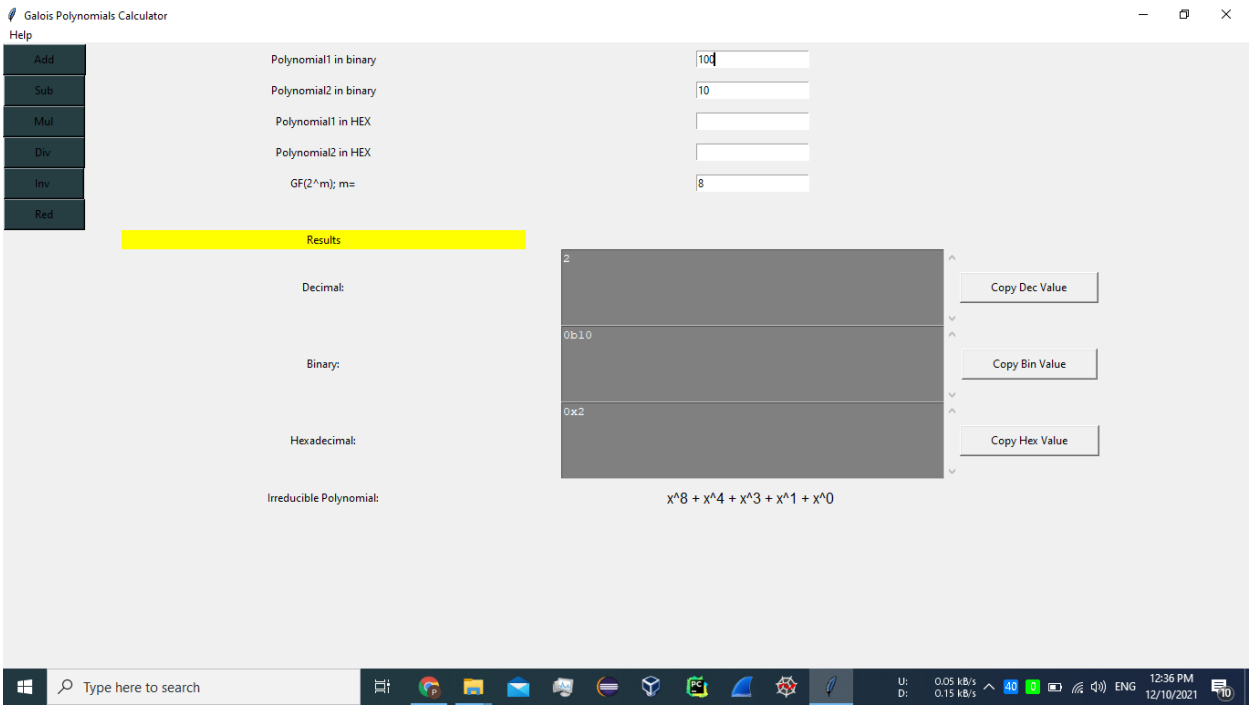
After checking the inputs and making sure everything is the way it should be, the operation function is called (depending on what button was pressed: add or sub or mul or…), if the inputs are outside the Galois field specified by the user, the inputs will be reduced and then the function called will be done. Using NumPy to operate on the polynomials and then reducing them appropriately we implemented each operation.

Some Screenshots of the app:

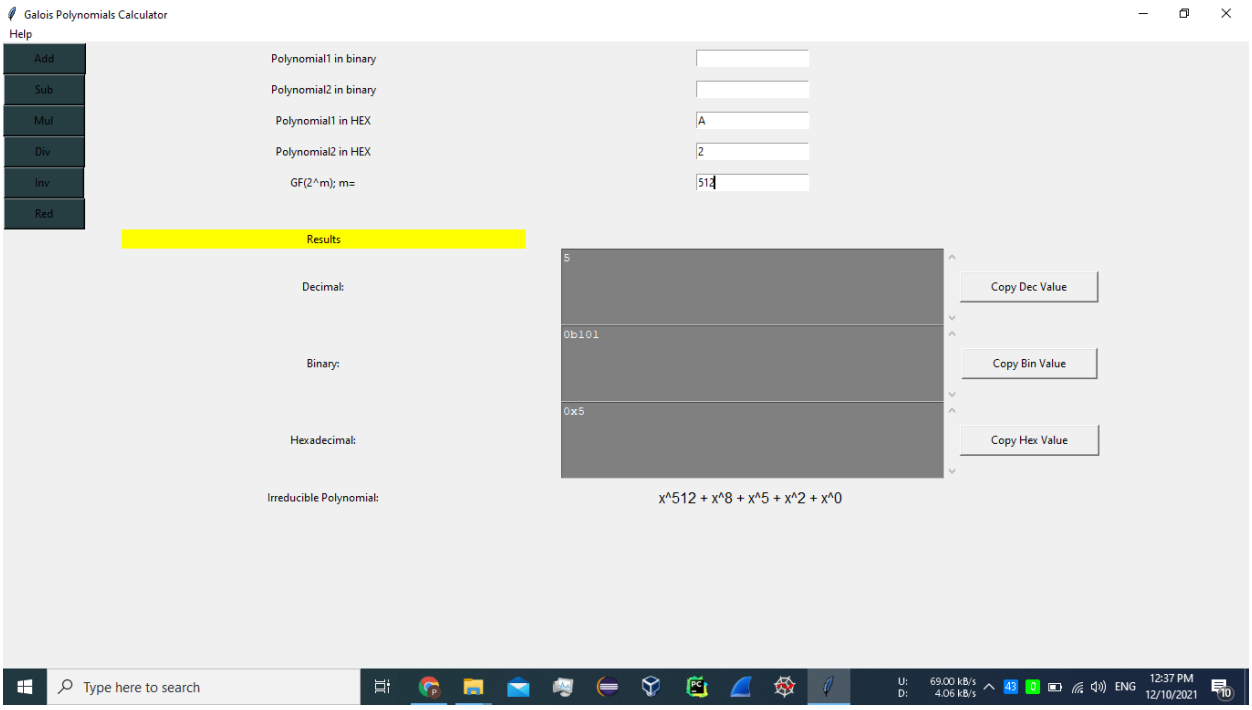
Addition:



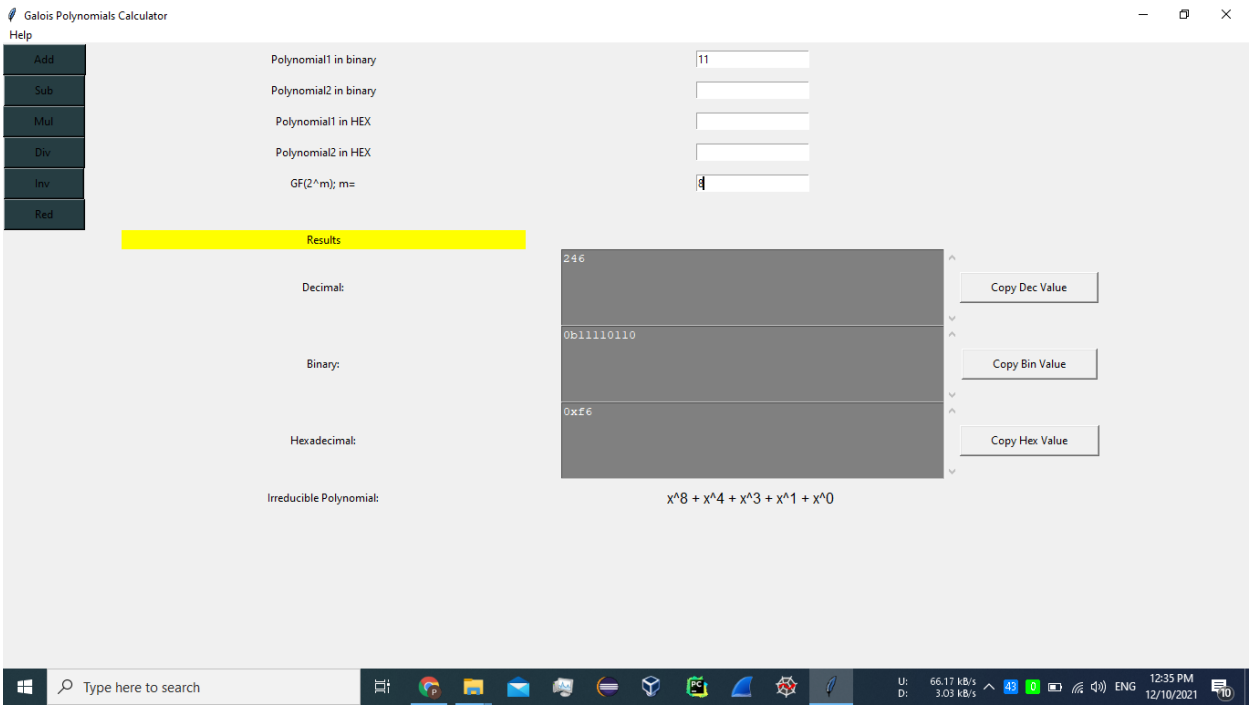
Division:



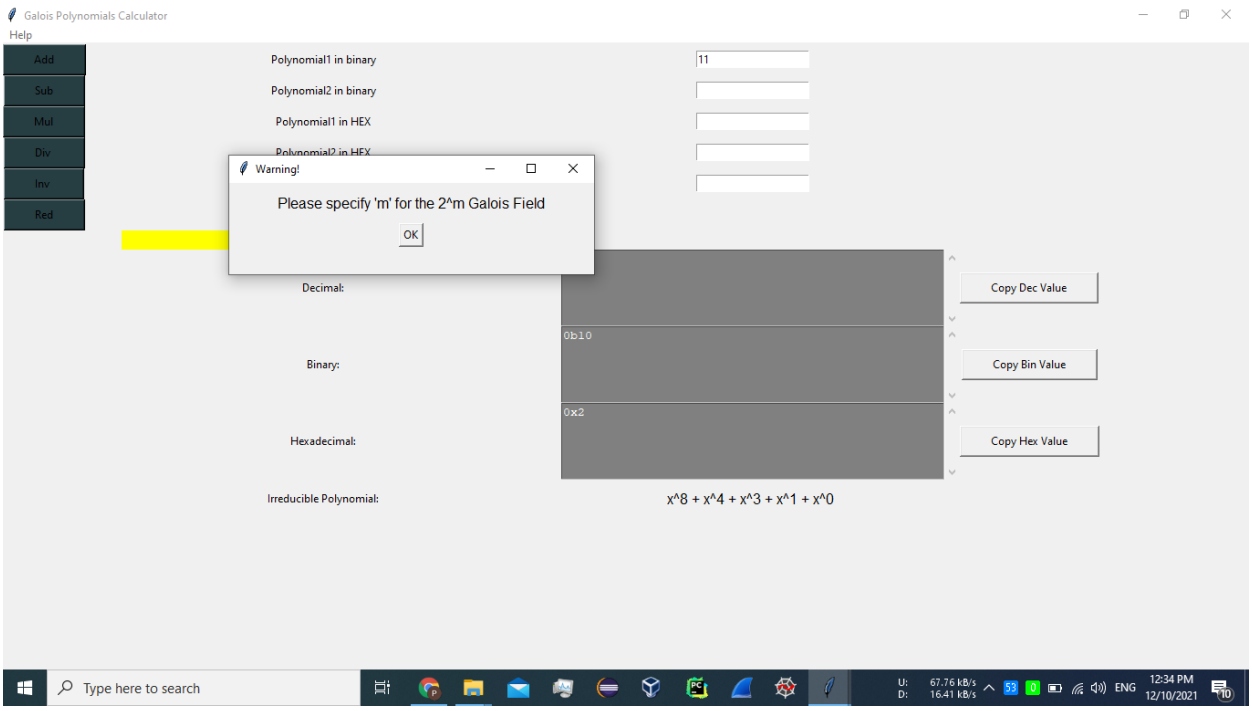
Division with hex input:



Inverse:


Galois Polynomials Calculator
Help
Add
Sub
Mul
Div
Inv
Red
Polynomial1 in binary
11
Polynomial2 in binary
Polynomial1 in HEX
Polynomial2 in HEX
GF(2^m); m=
8
Results
Decimal:
246
Copy Dec Value
Binary:
0b11110110
Copy Bin Value
Hexadecimal:
0xf6
Copy Hex Value
Irreducible Polynomial:
x^8 + x^4 + x^3 + x^1 + x^0
Type here to search
66.17 kB/s
3.03 kB/s
ENG
12:35 PM
12/10/2021

One warning:


Galois Polynomials Calculator
Help
Add
Sub
Mul
Div
Inv
Red
Polynomial1 in binary
11
Polynomial2 in binary
Polynomial1 in HEX
Warning!
Please specify 'm' for the 2^m Galois Field
OK
Decimal:
Copy Dec Value
Binary:
0b10
Copy Bin Value
Hexadecimal:
0x2
Copy Hex Value
Irreducible Polynomial:
x^8 + x^4 + x^3 + x^1 + x^0
Type here to search
67.76 kB/s
16.41 kB/s
ENG
12:34 PM
12/10/2021