**Turing Machine – Decide Invertibility**

**Collaborators:**

Owen Queen

Jared Staman

**Roles:**

Owen – Completed the red, yellow, and green portions of the Turing Machine (multiplication).

Jared – Completed the beige, orange, purple, and blue portions of the Turing Machine (subtraction/addition).

**Problem Description:**

Our project reads in a 2x2 matrix and determines if the matrix has an inverse or not. We knew we wanted to do something with matrices, and after some research and a talk with Dr. Berry, we decided that creating a Turing Machine that decided invertibility would be an adequate challenge. Our Turing Machine reads in unary numbers for a 2x2 matrix, along with their signs (+ or -), and performs multiplication and then subtraction to get the matrix’s determinant. If the determinant is non-zero, then the matrix is invertible; otherwise, when the determinant is zero, the matrix is not invertible.

Chart, scatter chart

Description automatically generated

**Function Description:**

1. Red – Sign detection
   1. This group of states determines, based on the first 4 characters, the signs (positive or negative) of the products of diagonal elements. This system simply looks at the sign inputs, classifying the product as negative if the signs are mismatched and positive if the signs match each other. The signs are written consecutively after the ‘$’ on the tape. The first one (left) is the sign for the product produced by the Yellow states, and the second one (right) is the sign for the product produced by the Green states (see below). The character ‘1’ denotes positive and ‘0’ denotes negative. After both signs are written, a ‘%’ is appended to the end of the tape.
2. Yellow – Multiplication of (0,0) and (1,1) diagonal elements
   1. These states perform multiplication on the elements at index (0,0) and (1,1) for the given matrix. The product of these two numbers is then written after the signs at the far-right region of the tape. When the product has been fully written, a ‘#’ is appended to the end of the tape.
3. Green – Multiplication of (0,1) and (1,0) diagonal elements
   1. Multiplication is then performed on the other diagonal elements not considered in the Yellow states. This algorithm is the exact same as the one for the Yellow states, writing the product after the ‘#’ that is at the end of the tape at the beginning of the program.
4. Beige – Addition or Subtraction
   1. The beige color transitions from the end of the multiplication into either the subtraction or addition parts. It goes to subtraction when either both signs are positive or both signs are negative. It goes to addition when one is positive and the other is negative.
5. Orange – Subtraction
   1. This part subtracts the two values that are given after the multiplication section. It returns either a sequence of 1’s or a 0. It then transitions to the blue section.
6. Purple – Addition
   1. When one number a positive number is subtracted by a negative number, addition occurs. This portion results in a 0 only when both numbers are zero. This section also transitions to the blue section.
7. Blue – Output
   1. Finally, after the addition/subtraction is done, the determinant is found. This part determines whether to output “invertible” or “not invertible”. When a zero is passed to this section, then “not invertible” is printed. When any other number is passed, then “invertible” is printed.

**How to Start TM:**

The format for the input tape is as follows:

S1 S2 S3 S4 % a # b # c # d $

Where:

1. S1, S2, S3, S4 are single numbers (0 or 1) denoting the sign on each entry. 0 is negative and 1 is positive.
2. % denotes the beginning of the matrix.
3. a, b, c, d are unary numbers consisting of a string of all 1’s of n size. That n value corresponds to the actual value in the matrix. For example, 4 would be 1111. The value 0 is just no 1’s.
4. # denotes the separation of individual elements of the matrix.
5. $ denotes the end of the matrix

Example 1:

If you wanted to test if the matrix [ (2, -1) , (-4, 3) ] was invertible, the input tape would be:

1001%11#1#1111#111$

The output for this would be:

1001%xx#x#1111#111 invertible

Example 2:

If you wanted to test if the matrix [ (0, 2), (0, -1) ] was invertible, the input tape would be :

0100%#11##1$

The output for this would be:

0100%#xx##1 not invertible