0. String Search (20 points)
Use the method given in class to design a linear-time algorithm to determine all occurrences of the string 10100101001 in strings over the alphabet \{0, 1\}. Submit your answer using the FSM website; (This isn’t quite up yet; enjoy your Thanksgiving!)

1. Diagonalization (30 points)
Let \(B\) be the set of all infinite binary sequences. Show that \(B\) is uncountable using a proof by diagonalization.

2. Countability (20 points)
An integer quadratic polynomial is a function of the form \(f(x) = ax^2 + bx + c\) where \(a \neq 0, b, c \in \mathbb{Z}\). Show that the set of all integer quadratic polynomials is countable.

3. Undecidability (30 points)
Consider the set \(\text{Identity}\):

\[
\text{Identity} = \{ \text{CODE(P)} : P \text{ is a Java program with domain and codomain } \mathbb{N}, \text{ and for all inputs } x, P(x) = x \}
\]

For example, the following programs all are not in \(\text{Identity}\):

```
1 public nat f(nat x) {
2     System.out.println("hello");
3     return x^2;
4 }
```

```
1 public void f(nat x) {
2     System.out.println("I have no output");
3 }
```

```
1 public nat f(nat x) {
2     if (x % 2 == 0) {
3         return 2x;
4     }
5     else {
6         return x;
7     }
8 }
```

While the following programs all are in \(\text{Identity}\):

```
1 public nat f(nat x) {
2     return x;
3 }
```

```
1 public nat f(nat x) {
2     return x;
3 }
```

```
1 public nat f(nat x) {
2     if (true) {
3         return x;
4     }
5     else {
6         return 5;
7     }
8 }
```

Show that \(\text{Identity}\) is undecidable using the fact that the Halting Problem is undecidable.