

## Problem 9.1

### Part a

We will need three tapes for this Turing Machine. The first tape will be a counter representing the next number that will be considered for primality. When the next number is requested, we will do the following:

1. Copy the current number from the first tape into the third tape
2. Find the last prime on the second tape, and for each zero in the current prime, remove a zero from the number on the third tape. When you reach the end of the prime, return to the head of that prime. Do this until one of the following cases is reached:
  - (a) The prime reaches its last bit when the number in the third tape equals zero. This means that the number is evenly divisible by the prime (i.e. the current number is not prime).
  - (b) The number reaches zero before the prime has reached its last bit. This means that the number is not evenly divisible by the prime. Increment the number in tape 1.
3. Repeat this process until either the number is evenly divisible by a prime on the second tape or until all primes are exhausted. If all primes are exhausted, then the number is prime and we can append it to the end of the second tape and increment the number on tape 1. Otherwise, we can simply increment the number on the first tape.

### Part b

Here is a rough layout of the states for this Turing Machine:

- $c_0$ : This state is used when copying the number in the first tape into the third tape.
- $c_1$ : This state is used when moving the head of the third tape back to the beginning of the number after copying.
- $d$ : Used when traversing a prime number and current number to determine if the current number gets zeroed out when the prime number ends.
- $p_0$ : This state represents when the current number reaches zero before reaching the end of the current prime number (i.e. the current number might be prime).
- $p_1$ : The state transitioned into when we've discovered that the current number is not evenly divisible by any current prime numbers (meaning that the current number is prime).

- $n$ : This state represents when the current number reaches zero when reaching the end of the current prime number (i.e. the current number is not prime)

## Problem 9.2

### Part a

X	Y	Z	$(x+y)((x+z)) + (x+z)$
T	T	T	F
T	T	F	T
T	F	T	F
T	F	F	T
F	T	T	T
F	T	F	T
F	F	T	T
F	F	F	T

### Part b

X	Y	Z	$(x+y) + ((x+z) + (x+z))$
T	T	T	T
T	T	F	T
T	F	T	T
T	F	F	T
F	T	T	T
F	T	F	T
F	F	T	T
F	F	F	T

### Part c

The approach of trying every possible combination **true/false** value in the expression does generalize to an unbounded number of variables. This is an incredibly inefficient algorithm that has a running time of  $O(2^n)$  where  $n$  is the number of variables in the expression.