# Homework 23

Joe Baker, Brett Schreiber, Brian Knotten

March 18, 2018

## 39

## 40

Alice and Bob both know a secret key $k$, a one time pad, which is the same bitlength as the message $m$. Carol does not know any information about this secret key, and her best course of action for determining a random bit of the key is to guess. So Carol has probability $1/2$ of guessing any bit of the key using a randomized polynomial algorithm $A$.

The xor operation has the property such that $m \oplus k = c$, where $c$ is the ciphertext, and $c \oplus k = m$. Moreover, the xor function is one-to-one, meaning that no other $k$ can derive $c$ from $m$ and vice versa.

Alice can encrypt her message $m$ using a one time pad $k$ to get $c$ using xor. Bob can similarly decrypt $c$ into $m$ using xor. Since no other $k$ can derive $m$, and since Carol cannot determine any bit of $k$ with probability greater than $1/2$, it follows that Carol cannot derive any bit of $m$ with the same probability greater than $1/2$. So Alice and Bob have computational security on $m$.

## 41

Assume $P = NP$, let $f : \{0,1\}^* \to \{0,1\}^*$ be a one-way function, and let $y$ be an output of $f$.
Let $A$ be a non-deterministic poly-time Turing Machine that tries every possible $x$ until $f(x) = y$ and then returns $x$ i.e. $A$ solves the problem of inverting $f$.

Because $A$ solves the problem of inverting a one-way function in poly-time, the problem is in $NP$. By our assumption $P = NP$, so there the problem is also in $P$ and exists an efficient algorithm for inverting one-way functions. Therefore one-way functions do not exist.