Assignment#3 Key
1. Check $F$ to be sure it’s legit

- $\text{GreaterThanZero}(F, 0) = \exp(F, 0) > 0$
  $\text{GreaterThanZero}(F, y+1) = \text{GreaterThanZero}(F, y) \&\& (\exp(F, y+1) > 0)$

- $N(F) = \exp(F, 0)$

- $\text{WellFormed}(F) = \text{GreaterThanZero}(F, 2*N(F)) \&\& (\exp(F, 2*N(F)+1) == 1) \&\& (\exp(F, 2*N(F)+2) == 1)$

- Really, really inefficient
  $\text{NoExtraPrimeFactors}(F) = \sim \exists 2*(N(F)+1)<z\leq F/2 [\exp(F, z) > 0]$

- $\text{AllGood}(F) = \text{WellFormed}(F) \&\& \text{NoExtraPrimeFactors}(F)$

- There are other approaches for $\text{NoExtraPrimeFactors}$
1. Alternate approach to NoExtraPrimeFactors

• $\text{ValueOf}_F(F, 0) = \text{prime}(0)^{\text{exp}(F, 0)}$
  $\text{ValueOf}_F(F, y+1) = \text{ValueOf}_F(F, y) \times (\text{prime}(y+1)^{\text{exp}(F, y+1)})$

• $\text{ExpectedValueOf}_F(F) = \text{ValueOf}_F(F, \text{exp}(F, 2*N(F)+1))$

• $\text{NoExtraPrimeFactors}(F) = \text{ExpectedValueOf}_F(F) == F$

• The idea here is that we compute the expected value of $F$ from $F$, by multiplying all its expected factors. If there is an extraneous factor (a prime factor $> p_{2n+2}$) then our actual value of $F$ will be larger than expected and that can be checked just by using equality or, if you prefer, $\text{ExpectedValueOf}_F(F) < F$
2. Show $S$ inf. rec. iff $S$ is the range of a monotonically increasing function

- Let $f_S(x+1) > f_S(x)$, and $\text{Range}(f_S(x)) = S$. $S$ is decided by the characteristic function
  $$\chi_S(x) = \exists y \leq x \left[ f_S(y) = x \right]$$
  The above works as $x$ must show up within the first $x+1$ numbers listed since $f_S$ is monotonically increasing.

- Let $S$ be infinite recursive. As $S$ is recursive, it has a characteristic function where $\chi_S(x)$ is true iff $x$ is in $S$.
  Define the monotonically increasing enumerating function $f_S(x)$ where
  $$f_S(0) = \mu x \left[ \chi_S(x) \right]$$
  $$f_S(y+1) = \mu x > f_S(y) \left[ \chi_S(x) \right]$$
  As required, this enumerates the elements of $S$ in order, low to high.
3. If $S$ is infinite re, then $S$ has an infinite recursive subset $R$

- Let $f_s$ be an algorithm where $S = \text{range}(f_s)$ is an infinite set
- Define the monotonically increasing function $f_R(x)$ by
  \[
  f_R(0) = f_S(0),
  f_R(y+1) = f_S(\mu x \ [ f_S(x) > f_R(y) ]) \]
  - The above is monotonically increasing because each iteration seeks a larger number and it will always succeed since $S$ is itself infinite and so has no largest value. Also, $R$ is clearly a subset of $S$ since each element is in the range of $f_S$.
- From #2, $R$ is infinite recursive as it is the range of a monotonically increasing algorithm $f_R$.
- Combining, $R$ is an infinite recursive subset of $S$, as was desired.