Name: ____________________

Pid: ____________________

1. (10 points) Let $S \subseteq \mathbb{N}$ be a nonempty set. Show that S is decidable iff there is a funciton $f : \mathbb{N} \to \mathbb{N}$ such that f is computable, f is nondecreasing, and $\text{Im } f = S$.

Solution: Let us assume that S is decidable and $\mathcal{A}$ decides S. Let $x_{\min}$ be the minimal element of S. Consider the following algorithm.

```
function F(n)
    if n < x_min then
        return x_min
    end if
    Let x ← n
    while ¬A(x) and x > x_min do
        x ← x − 1
    end while
    return x
end function
```

It is clear that this algorithm computes the total function $f : \mathbb{N} \to \mathbb{N}$ such that

$$f(x) = \begin{cases} x_{\min} & \text{if } x < x_{\min} \\ \max\{y \in S \ : \ y \leq x\} & \text{otherwise} \end{cases}.$$

Therefore f is nondecreasing. We need to prove now that $\text{Im } f = S$. To prove this firt we note that it is easy to see that $\text{Im } f \subseteq S$. In addtion, if $x \in S$, then $f(x) = x$, which implies that $\text{Im } f = S$.

Let us now prove the statement in the opposite dirrection. Assume that there is a total nondecreasing funciton $f : \mathbb{N} \to \mathbb{N}$ such that $\text{Im } f = S$.

- If S is finite, then we prove in class that it is decidable.
- Let S be an infinite set, and let $\mathcal{F}$ be an algorithm computing f. Consider the following algorithm

```
1: function A(x)
2:     while F(n) < x do
3:         n ← n + 1
4:     end while
5:     if F(n) = x then
6:         return 1
7:     else
8:         return 0
9:     end if
10: end function
```

  In lines 2-4 this algorithm finds the minimal n such that $f(n) \geq x$ (it always exists since S is infinite). If $x \in S$, then such a minumal n is equal x, and the algorithm returns 1. Otherwise $f(n) \neq x$ and the algorithm returns 0.

2. (10 points) Let $A, B \subseteq \mathbb{N}$ be enumeratable sets. Show that $A \times B$ is enumeratable.

Solution: Let $\mathcal{A}$ and $\mathcal{B}$ be semideciding algorithms for A and B, respectively. Consider the following algorithm.

```
function C(x, y)
    A(x)
    B(x)
    return 1
end function
```

Note that if $x \in A$ and $y \in B$, then the algorithm return 1. Otherwise the algorithm never terminates. Therefore, if $(x, y) \in A \times B$, then the algorithm return 1. Otherwise the algorithm never terminates.