Notes from March 29: analyzing Euclid's algorithm using induction

Consider the following algorithm, which takes two nonnegative integers a and b is inputs (note that $a\%b$ below denotes $a \bmod b$):

```
def Euclid(a, b):
    while b>0:
        a, b = b, a \% b
    return a
```

We already proved that this algorithm computes $\gcd(a, b)$ because $\gcd(a, b) = \gcd(b, a \bmod b)$ and because $\gcd(a, 0) = a$. We will now analyze how long it takes. For now, we will assume $a > b > 1$, so the proof is simpler; we will remove this assumption later.

Let $f_0 = 0$, $f_1 = 1$, $f_{n+2} = f_{n+1} + f_n$ for $n \geq 0$ be the Fibonacci sequence.

Theorem 1. *For any $n \geq 1$, if $a > b \geq 1$ and $\texttt{Euclid}(a, b)$ takes n iterations of the `while` loop, then $a \geq f_{n+2}$ and $b \geq f_{n+1}$.*

Proof. By induction of n. If $n = 1$ and $b \geq 1$, then $b \geq f_2$ (because $f_1 = 1$). Since $a > b$ and $b \geq 1$, then $a \geq 2$, and therefore $a \geq f_3$ because $f_3 = 2$.

Suppose, by inductive hypothesis, the statement is true for $n = k \geq 1$. We will prove the statement for $n = k + 1$. So suppose $\texttt{Euclid}(a, b)$ takes $k + 1$ iterations of the `while` loop. Then $\texttt{Euclid}(b, a\%b)$ takes k iterations of the while loop, because the first of the $k + 1$ iterations of $\texttt{Euclid}(a, b)$ will replace a with b and b with $a\%b$, and then there will be k iterations left. Since $k + 1 \geq 2$, $a\%b \geq 1$ (otherwise the loop would have stopped after one iteration). And by definition of remainder, $b > a\%b$. Therefore all the preconditions of the inductive hypothesis are satisfied (namely $\texttt{Euclid}(b, a\%b)$ takes k iterations and $b > a\%b \geq 1$). Thus, from the inductive hypothesis, we can conclude that $b \geq f_{k+2}$ and $a\%b \geq f_{k+1}$.

What's left to prove is that $a \geq f_{k+3}$. Indeed, $a = bq + a\%b$. Since $a > b$, we know $q \geq 1$ (else $bq \leq 0$ and $a\%b < b < a$, so $bq + a$ would be less than a). Therefore, $a = bq + a\%b \geq b \cdot +a\%b \geq f_{k+2} + f_{k+1} = f_{k+3}$. This concludes the proof. □

The theorem says that if Euclid's algorithm takes many iterations on some inputs, then those inputs must have been big. We will now flip the statement and say that if the inputs are not very big, then it can't take many iterations. We will state it as a "corollary" (we could have also used the word "theorem" here — the term "corollary" simply emphasizes that it's a very simple consequence of the previous result).

Corollary 1. *For any integer $n \geq 1$, if $a > b \geq 1$ and $b < f_{n+1}$, then $\texttt{Euclid}(a, b)$ takes fewer than n iterations.*

Proof. Supposes, for purposes of contradiction, that $\texttt{Euclid}(a, b)$ takes m iterations where $m \geq n$. Then, by the previous theorem, $b \geq f_{n+1}$. But this contradicts the assumption that $b < f_{n+1}$. □

The above corollary is known as Lamé's theorem.

Let us now remove the restriction that $a > b \geq 1$ and state results for any nonnegative a and b.

Corollary 2. *For any integer $n \geq 2$, if $a < f_n$ or $b < f_n$, $\texttt{Euclid}(a, b)$ takes fewer than n iterations.*

Proof. First, note that "$a < f_n$ or $b < f_n$" implies $\min(a, b) < f_n$.

We will proceed by cases.

It is easiest to rule out the cases with 0s first.

Case 1: $b = 0$. Then $\texttt{Euclid}(a, b)$ takes 0 iterations, which is fewer than n, because $n \geq 2$.

Case 2: $b > 0$ and $a = 0$. Then $\texttt{Euclid}(a, b)$ takes 1 iteration (because $a\%b = 0$), which is fewer than n, because $n \geq 2$.

Case 3: $b > 0$ and $a > 0$. We will divide this case further into three subcases.

Case 3a: $a > b$. Then "$a < f_n$ or $b < f_n$" implies $b < f_n$ (otherwise we'd have $a > b \geq f_n$). Since $a > b \geq 1$, we can apply the previous corollary to conclude that $\texttt{Euclid}(a, b)$ takes fewer than $n - 1$ iterations, which is fewer than n.

Case 3b: $a = b$. $\texttt{Euclid}(a, b)$ takes just one iteration (because $a\%b = 0$), so it takes fewer than n iterations because $n \geq 2$.

Case 3c: $b > a$. Then the first iteration will simply swap a and b, because $a\%b = a$. So after the first iteration, we are in case 3a, and $\texttt{Euclid}(a, b)$ will take fewer than $n - 1$ additional iterations, for total that is less than $1 + (n - 1) = n$.

□