

Homework 8

Problems

(1) (4pts.) Consider the following version of the Euclidean Algorithm to compute $\gcd(a, b)$:

Step 1: Swap the numbers if necessary to have $a \leq b$.

Step 2: If $a = 0$, then return b .

Step 3: If $a \neq 0$, then replace b by $b - a$ and go to **Step 1**.

(a) Carry out this algorithm to compute $\gcd(19, 2)$.

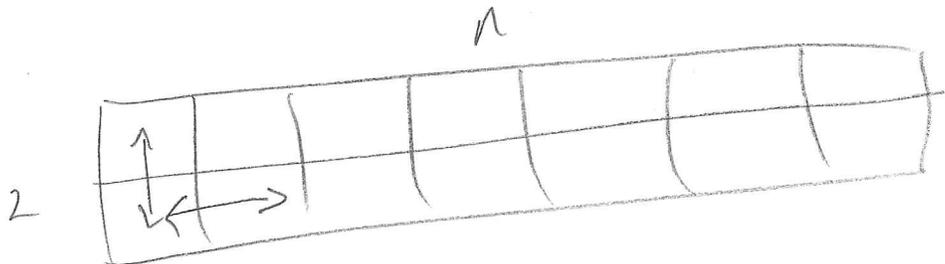
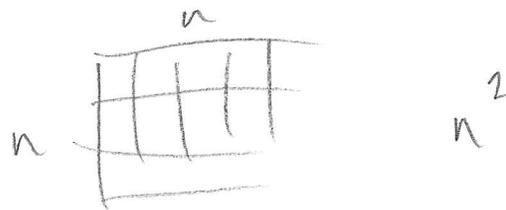
(b) Show that the modified Euclidean Algorithm always terminates with the right answer.

(c) How long does this algorithm take, in the worst case, when applied to two 100-digit integers?

(2) (3pts.) Prove that $F_0^2 + F_1^2 + \dots + F_n^2 = F_n F_{n+1}$.

(3) (3pts.) In how many ways can you cover a $2 \times n$ chessboard by dominoes?

$$\begin{aligned} &\gcd(F_k, F_{n+1}) \\ &(F_{k+1}, F_{n+2}) \\ &(F_{n+2}, F_{n+3}) \end{aligned}$$



$$n = (n-1) + (n-2)$$