Discrete Math Challenge HW 6 (2 points)

Write a Disco function implementing the *extended* Euclidean algorithm, which finds not only the GCD of a and b, but also integers s and t such that $sa + tb = \gcd(a, b)$. That is, define a function $egcd : \mathbb{N} \times \mathbb{N} \to \mathbb{Z} \times \mathbb{Z} \times \mathbb{N}$ such that if $egcd(a, b) = (s, t, g)$, then $sa + tb = g$ and g is the GCD of a and b. Some hints:

- Start by writing a recursive helper function

  $$egcdH : (\mathbb{Z} \times \mathbb{Z} \times \mathbb{N}) \times (\mathbb{Z} \times \mathbb{Z} \times \mathbb{N}) \to \mathbb{Z} \times \mathbb{Z} \times \mathbb{N}$$

  which takes the previous and current rows of the table and returns the last row of the table (the row containing the GCD of a and b).

- Then implement $egcd : \mathbb{N} \times \mathbb{N} \to \mathbb{Z} \times \mathbb{Z} \times \mathbb{N}$ simply by calling *egcdH* with the right values for the starting rows.