4330 Assignment 8 [*]

June 4, 2020

Write your code for the following problems in a single file named:

hw8-*lastname*.py

Consider the following Python implementation of the Euclidean Algorithm:

```
def gcd(a, b):
```

```
def gcd(a,b):
    if b==0:
        return a
    return gcd(b, a%b)
```

(1) (10 points) (On paper, separate from your code) Use this function by hand to calculate the gcd of 1266 and 888.

(2) (10 points) (On paper, separate from your code) Let u and v be positive integers. Show that if $u > v$ then $(u \bmod v) < u/2$.

(3) (10 points) (On paper, separate from your code) Use the previous result to prove that the Python function `gcd` above will never need more than $2 + 2\log_2 b$ divisions to compute the gcd of a and b. (In particular, the Euclidean Algorithm is polynomial-time in the input size $\log_2 a + \log_2 b$).

(4) (10 points) Write a function `gcd_divops`, by making a very small modification to the function `gcd`, so that it returns the number of divisions performed instead of the gcd itself.

DO NOT USE global variables, insert additional arguments to the function, or use any Python magic that we haven't learned about; this can and should be done using only ideas and concepts that have been covered in class. The resulting function should still be recursive and look almost identical to the original function, except that it returns the number of divisions instead of the gcd.

(5) (10 points) Add code to your program to determine the average number of divisions performed when using the above function to compute the gcd of two integers $a, b \in [1, 10]$. Then repeat it, but for $a, b \in [1, 100]$, and again for $a, b \in [1, 1000]$.