Fall 2023 CIS 3362 Homework #5 Grading Criteria

1) 5 pts – give full credit if correct, subtract 1 pt for each error, cap at 0.

2) 5 pts – give full credit if correct, subtract 1 pt for each error, cap at 0.

3) 5 pts – 1 pt stating Fermat, 2 pts to rewrite exponent, 2 pts to get to correct answer.

4) 5 pts – 2 pts phi calculation, 2 pts to rewrite exponent, 1 pt final answer.

5) 10 pts – 2 pts stating that k = 6 and m = 27 from step 1.
3 pts – correct calculation for 2^{27} = 645 mod 1729 shown
3 pts - correct calculation for 645^2 = 1065 mod 1729
2 pts - correct calculation for 1065^2 = 1 mod 1729 and concluding it's composite

6) 10 pts – 2 pts for starting chart at x = 213.
5 pts for the rows on the chart, ½ pt off round down if a row is incorrect
3 pts to use last row of chart to get a factorization (just 161 x 281 is needed)

7) 10 pts – 8 pts for proving that if gcd(k, p – 1) = 1, then α^k is also a primitive root
8 pt breakdown:
- 1 pt – starting with assume primitive root alpha (or whatever)
- 2 pts – considering raising alpha to a power k as described above
- 2 pts – looking at exponents of alpha
- 3 pts - completing proof by contradiction

2 pts for proving if gcd(k, p – 1) != 1, then α^k is not a primitive root.

Note: other ways to do this, so if you aren't sure let me grade it.

8) 10 pts - 4 pts for making the observation that each base is counted exactly once
In the sum on the left without proof.
6 pts for proving that if gcd(k, p – 1) = d, then the order of α^k is $\frac{p-1}{d}$.

Note: other ways to do this, so if you aren't sure let me grade it.

9) 40 pts – There are two test cases cycle_small.in and cycle_large.in
Automatic 20/40 if cycle_small works but cycle_large doesn't.
Automatic 40/40 if cycle_large works.
Time limit = 2 seconds

Partial credit for incorrect solutions:

If a quick fix fixes either case, then just take off for the error what

you think is appropriate.

If there's an overflow error affecting answers: -5

Partial positive credit for slow solution
2 pts for processing cases in main
3 pts for looping through all bases
5 pts for trying to exponentiate each base
5 pts for keeping a counter and adding to it

Partial positive credit for fast solution
5 pts – attempted own phi function
5 pts – identified each prime divisor in phi function
5 pts – tried to use each prime divisor appropriately in phi function
10 pts – theoretically summing up correct terms (d*phi(d) for appropriate d)
5 pts – runs to only the square root.