Homework 5: Number Theory

Due date: Wednesday October 30th at 11:59 PM
Version 3: Updated 10/24 9 PM. Corrected graph in question 7b.
If you work with others (and you should!), remember to follow the collaboration policy outlined in the syllabus.
In general, you are graded on both the clarity and accuracy of your work. Your solution should be clear enough that someone in the class who had not seen the problem before would understand it.
We sometimes describe approximately how long our explanations are. These are intended to help you understand approximately how much detail we are expecting. You are allowed to have longer explanations, but explanations significantly longer than necessary may receive deductions.

Be sure to read the grading guidelines on the assignments page for more information on what we're looking for.

In order to assist with the transition from formal proofs to English proofs, we've published a style guide on the website containing some tips. This guide contains references to proof materials that we haven't taught yet, so don't worry if some of these terms are unfamiliar.

1. Backwards Proofs [6 points]

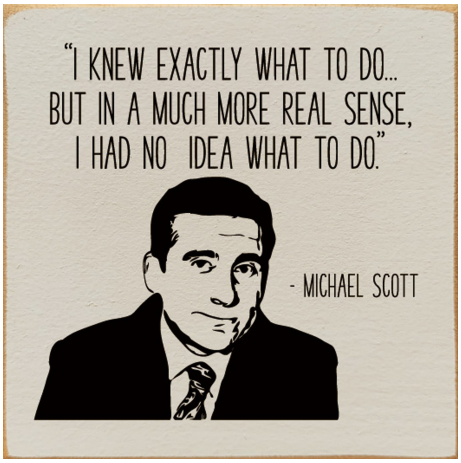
A common error now that we're doing a lot of algebra is to write a "backwards" or "U-shaped" proof. For a proof to be valid, we must start from facts we know (either givens, or accepted facts, or supposing hypotheses to prove implications), and derive from them the statement we desire.
We cannot start from the equation to be shown and simplify it to something "obviously true."

Read the reading about backwards proofs. Then do these problems:

(a) Complete the Backwards Proof Practice assignment on Gradescope. This part will feel like a concept check (explanations appear when correct, problem is graded automatically, etc.) but counts in the homework category for grades. [4 points]

(b) Help correct your friend's first two steps of the proof if it was wrong or finish it up for them if it was right! [2 points]

2. Let's Settle Our Differences [5 points]



The characters in this problem are based on The Office.

Congratulations! You have been promoted to Assistant to the Regional Manager at Dunder Mifflin (don't tell Dwight!). As your first task as Assistant to the Regional Manager you have three sets of paper: A, B, and C. Jim has told you that the relation between the sets of paper is $(A \cup B) \setminus C$. Michael believes the relation between the sets of paper is $(A \setminus C) \cup (B \setminus C)$. Pam suggests you show Michael that Jim's statement is a subset to Michael's, but warns you that Michael does not have the patience to read a full formal proof.

Your job: Suppose A, B, and C are sets. Prove in English $(A \cup B) \setminus C \subseteq (A \setminus C) \cup (B \setminus C)$.

3. Like -5 but better! [22 points]

In normal arithmetic, $a + 5 + (-5) = a$ for every integer a. So we say that -5 "undoes" 5. In modular arithmetic, a similar statement might be that $a+5+2 \equiv a \pmod 7$, so 2 "undoes 5 for $\pmod 7$ addition." More generally, given an integer n, we say that an integer b "undoes 5 for $\pmod n$ addition" if and only if for all integers a, $a+5+b \equiv a \pmod n$.

In this problem, you will show that for every integer n (where $n > 5$), there exists some integer b, where $1 \leq b \leq n$, which undoes 5 for $\pmod n$ addition.

(a) Write the statement "for every integer n (where $n > 5$), there exists some integer b, where $1 \leq b \leq n$, which undoes 5 for $\pmod n$ addition." in predicate logic. You should use the predicate "$\texttt{Undoes5}(b, n)$ to say "b undoes 5 in $\pmod n$ arithmetic". [2 points]

(b) You (hopefully!) have a statement which starts $\forall n \exists b$. Recall that since the $\exists$ come second, the value of b is allowed to depend on n. Give a formula for the b (in range $1 \leq b \leq n$) for which $\texttt{Undoes5}(b, n)$ evaluates to true. The formula will depend on n. [2 points]

(c) Now do the actual proof. You'll start the proof by introducing an arbitrary variable (you're proving a $\forall$) then you'll be doing an exists proof (tell us what value of b you want and argue that it makes $\texttt{Undoes5}()$ evaluate to true). Be sure you don't do a backwards proof! **For this part, you may not use facts about modular equivalences**, you may only use the definitions of divides and equivalence mod n and algebra (applied to equations or individual numbers, not to equivalences). [8 points]
Hint: Don't forget that the definition of `Undoes5` has another $\forall$ quantifier inside of it!

Now that we've shown there is a way to undo 5, next we're going to try to show there's not a bunch of different ways. In this problem, you'll show that for every integer n (where $n > 5$), for all integers b, b' where both b and b' undo 5 for $\pmod n$ addition, that $b \equiv b' \pmod n$. Note that we've gotten rid of the $1 \leq b \leq n$ requirement in this part! [6 points]

(d) Write the statement above in predicate logic. Use the predicate $\texttt{Undoes5}(b, n)$ for "b undoes 5 for $\pmod n$ arithmetic." [2 points]

(e) Now write an English proof of the statement. For this part you may use theorems shown in class and on the Number Theory Reference Sheet. [8 points]
You may also use the following fact:

Theorem 1 (Transitivity of Equivalence)**.** *For all integers a, b, c, n with $n > 0$:*
if $a \equiv b \pmod n$ and $b \equiv c \pmod n$ then $a \equiv c \pmod n$.

(f) For similar concepts in modular arithmetic, people will say things like "There is a unique number that undoes $5 \pmod n$." Ponder why this use of "unique" makes sense, but also why this is a little different from the example of "unique" we saw in class. You do not have to write anything for this part [0 points]

4. GCD Proof [12 points]

Bezout's Theorem tells us that if x and y are positive integers, then there exist some integers a and b such that

$\gcd(x, y) = ax + by$. However, the converse isn't always true: there could exist some integers a and b such that $d = ax + by$, but d isn't necessarily $\gcd(x, y)$. In this problem, we will see a special case where the converse does hold.

(a) For all **positive** integers x and y, prove the following claim: if there exist some integers a and b such that $ax + by = 1$, then $\gcd(x, y) = 1$. [9 points]

You may use without proof that if any integer k satisfies $k|1$, then k must be either 1 or -1. **Hint:** The facts about GCD that you will need for this problem are that if $d = \gcd(x, y)$ then $d|x$ and $d|y$, and it is the largest integer that does this.

(b) Use part (a) to show that $\gcd(m, m + 1) = 1$ for all positive integers m. [3 points]

5. Quartic Modulo [8 points]

Prove that for all integers, a, b, and n, where $n > 0$: if $a \equiv b \pmod{n}$, then $a^4 \equiv b^4 \pmod{n}$. For this problem, you may not use facts proven in lecture (e.g., you may not use that $a \equiv b \pmod{n} \rightarrow ac \equiv bc \pmod{n}$). You can use the definitions of modular equivalence and divides, and do any algebra you like on regular equations.
Hint: Our first algebra step involves multiplying by $b+a$. We suggest basing future algebra steps off that one.

6. A Proof By Contradiction [7 points]

Use a proof by contradiction to show the following claim: for all integers n, if $n \equiv 1 \pmod{6}$, then $n \not\equiv 2 \pmod{4}$.
Hint: You may use without proof that a number cannot be both even and odd.

7. Counterexamples Galore [6 points]

In this problem you will use proof by counterexample to disprove claims in a wider set of problems than what we've seen in class. For each part, provide a counterexample that disproves the given claim. Remember to provide one counter-example, not a class of them.

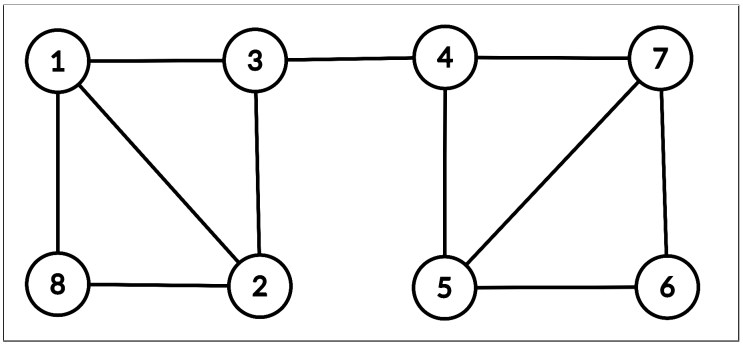
(a) **Buggy Algorithm**: Your friend claims "For every linked list whose nodes have distinct integer values, the function below will remove the node with the target value if such a node exists in the list." Disprove this claim with a counter-example.

For your counterexample, provide a LinkedList and a target value. Briefly explain why the code fails on your example.

```
public void removeNode(ListNode head, int target) {
    ListNode curr = head;
    while(curr.next != null) {
        if (curr.next.val != target) {
            curr = curr.next;
        } else {
            curr.next = curr.next.next;
        }
    }
}
```

(b) *You may have seen graphs like the one below in an introductory programming course. In case you haven't: we call the circles in the graph "vertices" and the lines connecting two vertices "edges". No further understanding of graphs is required to complete this problem.*

Husky Edge Coloring: Your friend claims "There exists no way to color the edges of the graph below using colors purple, gold, and pink such that every vertex is connected to at most one edge of each color [for your counterexample, provide a coloring of the edges of the graph].
For this part you do not have to explain your counter-example.



8. Feedback [1 point]

Answer these questions on the separate gradescope box for this question.

Please keep track of how much time you spend on this homework and answer the following questions. This can help us calibrate future assignments and future iterations of the course, and can help you identify which areas are most challenging for you.

- How many hours did you spend working on this assignment (excluding any extra credit questions, if applicable)? Report your estimate to the nearest hour.
- Which problem did you spend the most time on?
- Any other feedback for us?

9. Extra Credit: Exponentially increasing fun [0 points]

You will submit this question to the separate gradescope box for "homework 5 extra credit."

Since $a \equiv a \ \% \ n \pmod{n}$, we know that we can reduce the base of an exponent in $\pmod{n}$ arithmetic. That is:

$$a^k \equiv (a \ \% \ n)^k \pmod{n}.$$

But the same is **not** true of the exponent! That is, we cannot say that $a^k \equiv a^{(k \ \% \ n)} \pmod{n}$. Consider, for instance, that $2^{10} \equiv 1 \pmod{3}$ but $2^{(10 \ \% \ 3)} \equiv 2 \pmod{3}$. The correct way to simplify exponents is quite a bit more subtle. In this problem you'll prove it in steps.

For these proofs you may use any theorem on the number theory reference sheet, even the ones we haven't proven yet in class.

(a) Let $R = \{t \in \mathbb{Z} : 1 \leq t \leq n-1 \wedge \gcd(t, n) = 1\}$. Define the set $aR = \{(ax) \ \% \ n : x \in R\}$. Prove that $aR = R$ for every integer $a > 0$ with $\gcd(a, n) = 1$.

(b) Consider the product of all elements in R (taken $\pmod{n}$) and consider the product of all the elements in aR (again, taken $\pmod{n}$). By comparing these two expressions, conclude that for all $a \in R$ we have $a^{\varphi(n)} \equiv 1 \pmod{n}$ where $\varphi(n) = |R|$.

(c) Use the previous part to show that for any $b \geq 0$ and $a \in R$ we have $a^b \equiv a^{(b \,\%\, \varphi(n))} \pmod{n}$.

(d) Now suppose that $y = x^e \,\%\, n$ for some x with $\gcd(x, n) = 1$ and e some integer ≥ 0 such that $\gcd(e, \varphi(n)) = 1$. Let $d = e^{-1} \,\%\, \varphi(n)$. Prove that $y^d \equiv x \pmod{n}$.

(e) Prove the following two facts about φ: First, if p is prime then $\varphi(p) = p - 1$. Second, for any positive integers a and b with $\gcd(a, b) = 1$, we have $\varphi(ab) = \varphi(a)\varphi(b)$.

These facts together are the basis for the most-widely used "public key encryption system." One chooses $n = pq$ for large primes p and q, and a value of e. The numbers n and e are made public to anyone who wants to send a message securely. To send a message x, the sender computes $y = x^e \,\%\, n$ and sends y (the "encrypted text"). To decrypt, one computes $y^d \,\%\, n$ (note that the recipient must be the one who chose p, q so they can calculate d). The security of the system relies on it being hard to compute d from just e and m.