CSC165H1: Problem Set 2

Due October 23 2019 before 4 pm

General instructions

Please read the following instructions carefully before starting the problem set. They contain important information about general problem set expectations, problem set submission instructions, and reminders of course policies.

- Your problem sets are graded on both correctness and clarity of communication. Solutions which are technically correct but poorly written will not receive full marks. Please read over your solutions carefully before submitting them. Proofs should have headers and bodies in the form described in the course note.
- Each problem set may be completed in groups of up to three. If you are working in a group for this problem set, please consult `https://github.com/MarkUsProject/Markus/wiki/Student_Groups` for a brief explanation of how to create a group on MarkUs.

  Exception: Problem Sets 0 and 1 must be completed individually.
- Solutions must be typeset electronically, and submitted as a PDF with the correct filename. **Handwritten submissions will receive a grade of ZERO.**

  The required filename for this problem set is **problem_set2.pdf**.
- Problem sets must be submitted online through MarkUs. If you haven't used MarkUs before, give yourself plenty of time to figure it out, and ask for help if you need it! If you are working with a partner, you must form a group on MarkUs, and make one submission per group. "I didn't know how to use MarkUs" is not a valid excuse for submitting late work.
- Your submitted file should not be larger than 9MB. This may happen if you are using a word processing software like Microsoft Word; if it does, you should look into PDF compression tools to make your PDF smaller, although please make sure that your PDF is still legible before submitting!
- The work you submit for credit must be your own; you may not refer to or copy from the work of other groups, or external sources like websites or textbooks. You may, however, refer to any text from the Course Notes (or posted lecture notes), except when explicitly asked not to.

Additional instructions

- For each proof you write, make sure to first write in predicate logic the precise statement, fully simplified, that you're going to prove. For a disproof, clearly write the fully simplified negation. You do **not** need to show your work for computing negations of statements. Your proof should include a **header**, where names and assumptions are introduced, and a **body**, where statements leading to the conclusion are derived.
- You may use external facts only from the course notes, worksheets, or lecture. If you use one of these, you must let the reader know what external fact you are using and where you got it.

- For any *concrete numbers*, you may state whether one divides another, or whether a number is prime, without proof. For example, you can write statements "3 | 12" and "15 is not prime" without justification.

1. **[9 marks] mod 3** Prove each of the following statements. You may use the Quotient Remainder Theorem from the course notes.

   (a) **[3 marks]** Every integer n satisfies either $9 \mid n^2$ or $3 \mid n^2 - 1$.

   (b) **[3 marks]** No integer n satisfies $n^2 \equiv 2 \pmod 3$.

   (c) **[3 marks]** Every integer n satisfies either $n^2 \equiv 0 \pmod 4$ or $n^2 \equiv 1 \pmod 4$.

2. **[6 marks] more congruence.**

   (a) **[3 marks]** Prove that if p_1 and p_2 are distinct primes, and a and b are any integers, then

   $$a \equiv b \pmod{p_1 p_2} \Leftrightarrow a \equiv b \pmod{p_1} \wedge a \equiv b \pmod{p_2}$$

   (b) **[3 marks]** Prove that if p_1 and p_2 are distinct primes, and a and b are any integers, there exists exactly one integer x that satisfies:

   $$x \equiv a \pmod{p_1} \wedge x \equiv b \pmod{p_2} \wedge 0 \leq x \wedge x < p_1 p_2$$

   Hint: If two different integers x_1 and x_2 satisfy the first two conditions, what can you prove about their remainder $\pmod{p_1 p_2}$?

3. **[9 marks] alternations...**

   For each of the following statements, decide whether you believe it is true or false. If you believe it is true, prove the statement. If you believe it is false, negate the statement and prove the negation.

   You need to approach each part seriously. There are no marks for "proving" a false statement true, or "proving" a true statement false.

   (a) **[3 marks]**

   $$\forall e \in \mathbb{R}^+, \exists d \in \mathbb{R}^+, \forall x \in \mathbb{R}, |x| < d \Rightarrow |7x| < e$$

   (b) **[3 marks]**

   $$\exists d \in \mathbb{R}^+, \forall e \in \mathbb{R}^+, \forall x \in \mathbb{R}, |x| < d \Rightarrow |7x| < e$$

   (c) **[3 marks]**

   $$\forall d \in \mathbb{R}^+, \exists e \in \mathbb{R}^+, \forall x \in \mathbb{R}, |x| < d \Rightarrow |7x| < e$$

4. **[6 marks] a prime example...**

   Euclid proved that there are infinitely many prime numbers, and this is the approach used in the course notes Theorem 2.3. In the questions below you may use Exercise 2.19 on modular arithmetic, and the divisibility of linear combinations.

(a) **[3 marks]** Prove there are infinitely many primes congruent to 5 (mod 6). **Hint:** Think about the **technique** of Theorem 2.3 in the course notes, and also that there are other arithmetic manipulations other than adding one, such as multiplying by 5.

(b) **[3 marks]** Prove or disprove that for any natural number n there is a natural number m that is not prime, is larger than n, and with $m \equiv 5 \pmod 6$.

5. **[3 marks]** **subsets** If a set S has n elements, how many subsets of size 4 does it have? Investigate this for sets of size 0, 1, 2, 3, 4, and 5, then make a conjecture. Prove your conjecture using simple induction. You may **not** use any external facts or techniques from combinatorics, but you **may** assume (without proof) the fact that a set with n elements has $n(n-1)(n-2)/6$ subsets of size 3.

6. **[3 marks]** **number representation** Read Theorem 4.1 carefully. It claims there is at least one binary representation for any natural number. The base case explicitly gives one representation for the natural number 0. Trace through the proof to see what binary representation it guarantees for the natural number 5. What is the representation? Explain.