MTH 225: Checkpoint 10

Learning Targets on this Checkpoint: 2, 3, 6, 7, 8, 9, 12, 13, and 14.

The first eight are the eight Core Learning Targets and are due in class. Learning Target 14 is non-Core and can be done outside of class, but must be submitted to Blackboard by 11:59pm ET today, November 15.

Instructions:

- Do not attempt any problem for which one or both of the following is true;
  - You **aren't ready to take it** because you need more practice; or
  - You **don't need to take it** beause you are already at Level 2 (or at Level 1 and you don't intend to go to Level 2).
- **A basic scientific or graphing calculator is the only technology allowed**. If you are using a smartphone app as your calculator, your phone *must* be switched to airplane mode. The use of any other technology, or phone apps with internet enabled, will be treated as academic dishonesty.
- **You are not allowed any notes** on Checkpoints.
- Make sure to **put your name on each page you are handing in.**
- Note that each problem has *Success Criteria* listed. These are the standards for what constitutes a "successful" demonstration of skill. Make sure to read through those criteria and double check your work against them before turning in your work.

Learning Target 2

> (**CORE**) I can add, subtract, multiply, and divide positive integers in base 2.

1. Add the base-2 integers `11110011` and `10000011`. **Show your work and circle your answer**.
2. Subtract the base-2 integers `11110011` and `10000011`. **Show your work and circle your answer**.
3. Multiply the base-2 integers `1110` and `110`. **Show your work and circle your answer**.
4. Divide the base-2 integer `1101100` by `11`. **Show your work and circle your answers**.

Success criteria: All four answers are correct, and the work leading to each answer is clear and legible. **All arithmetic must be done entirely in base 2 without converting to base 10 first.** Up to two "simple errors" (as described in the *Standards for Student Work* document) are allowed.

Learning Target 3

> (**CORE**) I can identify the hypothesis and conclusion of a conditional statement and state its converse, contrapositive, inverse, and negation.

Consider the statement: ***If the input is greater than 5, print an error statement.*** State, in clear English:

1. The hypothesis of the statement
2. The conclusion of the statement
3. The converse of the statement
4. The contrapositive of the statement
5. The inverse of the statement
6. The negation of the statement (Do not just put "not" or "it is not the case that" in front of the statement)

Success criteria: All six statements are given correctly. (There is no "work" required on this one.)

Learning Target 6

> (**CORE**) I can determine elements of a recursively-defined sequence using a recurrence relation and derive a recurrence relation for a recursively-defined sequence.

1. Consider the sequence of numbers a_n given by the recursive formula $a_0 = 2$, $a_1 = 1$, and for $n > 1$, $a_n = a_{n-1} - 2a_{n-2}$. Find the values of a_2, a_3, a_4 and a_5. Show your work on each calculation.
2. Consider the sequence of numbers: $6, 12, 24, 48, \ldots$. If $a_0 = 3$, give a recurrence relation that generates the rest of the sequence. (Do *not* give a closed formula, but a recursive definition.)

Success criteria: On the first part, the recursive formula must be used correctly with the correct base cases, and the mathematical work leading to the answers must be shown. On the second part, the recurrence relation must be recursive (not a closed formula) and correctly generate the terms of the sequence. No "work" is required on the second part. No more than two simple errors are allowed total.

Learning Target 7

> (**CORE**) Given a statement to prove using mathematical induction, I can state the framework of a proof. (Identify the predicate, identify and prove the base case, state the inductive hypothesis, and state what needs to be proven in the inductive step)

Suppose we want to prove the following statement using mathematical induction:

> For every integer $n \geq 1$, $1 + 2 + 4 + 8 + \cdots + 2^n = 2^{n+1} - 1$.

1. Write out the predicate involved in this statement: $P(n)$ says...
2. What value of n corresponds to the base case?
3. Prove that the base case holds, by direct computation or demonstration.
4. Clearly state the inductive hypothesis without using the notation "$P(n)$".
5. Clearly state what needs to be proven in the inductive step. Do not use the notation "$P(n)$" . You do NOT have to actually prove this statement; just state it.

Success criteria: The predicate is correctly identified and does not include a quantifier. The correct value of n for the base case is identified. The proof of the base case is clearly written and is correct. The inductive hypothesis is correctly and clearly stated and does not include "$P(n)$". The statement to be proven in the inductive step is correctly and clearly stated and does not include " $P(n)$".

Learning Target 8

> (**CORE**) I can convert a set given in roster notation to set-builder notation and vice versa; and determine if an object is an element of the set.

1. Here are three sets written in set-builder notation. For each, if the set is written using correct set syntax, rewrite it using roster notation. If the set is written using incorrect syntax, write **INCORRECT SYNTAX**. *See the success criteria below for what constitutes correct roster notation.*

   a) $\{x \in \{10, 11, 12, \ldots, 30\} \,:\, x \text{ is a power of } 2\}$

b) $\{x \in \mathbb{N} \, : \, x \,\%\, 3\}$

c) $\{2^n \, : \, n \in \mathbb{N}\}$

2. Rewrite the set $S = \{August, September, October, November, December\}$ using set-builder notation. There is more than one way to do this correctly; your answer must equal S and must use correct set notation and syntax. You may not use "$\{x \, : x \in S\}$" as an answer.

Success criteria: On the first part, all sets with incorrect syntax must be indicated; sets with correct syntax must:

- Have ALL its elements listed if there are 4 or fewer elements;
- Have AT LEAST FOUR elements listed if there are 5 or more elements;
- If the set is infinite, the roster notation must reflect this and not give a finite roster like {1,2,3,4}
- If the set is finite, the roster notation must reflect this and not give an infinite roster like {1,2,3,4,...}
- Have no more than two errors in the elements

On the second part, the set must equal S and the syntax of the set notation must be correct and rephrase the set in a meaningful way.

Learning Target 9

> (**CORE**) I can find the intersection, union, difference, symmetric difference, power set, cardinality, cartesian product, and complement of sets.

Let $A = \{1, 2, 4, 8\}$, $B = \{2, 4, 6, 8\}$, and $C = \{0, 5\}$ and suppose the universal set is $U = \{0, 1, 2, \ldots, 10\}$. Find each of the following. You do not need to show intermediate steps, but this is often helpful for you (and for the person grading your work).

1. $A \cap C$
2. $B \cup C$
3. $B \setminus A$
4. $\overline{A}$
5. $B \setminus (A \cap C)$
6. $((A \cap B) \cup (C \cap B)) \cap \emptyset$
7. $|\mathcal{P}(C)|$
8. $|A \times C|$

Success criteria: The final two items are correct; and at least 5 of the first 6 are correct.

Learning Target 12

> **(CORE)** I can apply the Additive and Multiplicative Principles and the Principle of Inclusion/Exclusion to formulate and solve basic combinatorics problems.

Find the following and show your steps:

1. In a school, there are 60 students. 25 students play basketball, 30 students play soccer, and 20 students play both basketball and soccer. Determine the number of students who play either basketball or soccer.

2. Suppose you are designing a password for a new computer system. The password must be a 6-character string using only digits (0-9) and lowercase letters (a-z). Additionally, the first character must be a digit, the second character must be a vowel (a, e, i, o, or u), and the remaining four characters can be any digit or lowercase letter. How many passwords are possible?

Success criteria: All items use a correct method for counting; correct computations are set up and shown clearly; and all answers are correct except for no more than two simple errors.

Learning Target 13

> **(CORE)** I can compute factorials and binomial coefficients, and apply the these concepts to solve basic combinatorics problems. (permutations, selections, distributions)

1. Compute the values of each of the following, and show your steps:

   (a) $9!$

   (b) $\binom{20}{18}$

2. When giving out candy at Halloween a coiuple of weeks ago, I had 20 packets of peanut M&M's to hand out, and was visited by a group of 5 kids. How many different ways can the candy be given to the five kids? You don't need to "explain" your reasoning (unless it helps you think) but make sure to show the formula you are using to get the answer as well as the numerical answer itself.

Success criteria: Both parts of the first item are correct except for no more than one simple error. On the second part, a correct method for counting is used, correct computations are set up and shown clearly, and the answers are correct except for no more than one simple error.

🆕 Learning Target 14

> I can determine if a sequence of numbers is arithmetic or geometric and derive both closed-form and recursive formulas for them.

Below are four integer sequences. Label each one as **arithmetic**, **geometric**, or **neither** (if the sequence is neither arithmetic nor geometric). If a sequence is either arithmetic or geometric. give both a closed formula and a complete recursive definition for the sequence. You do not need to show your work but your answers must be complete and correct. Assume that the starting index is 0.

1. $4, 6, 9, 13.5, 20.25, \ldots$
2. $1, 5, 9, 13, 17, \ldots$
3. $2, 5, 7, 12, 19, \ldots$
4. $4, 8, 12, 16, 20, 24, \ldots$

Success criteria: All four sequences are correctly labelled. For sequences that are either geometric or arithmetic, a correct closed formula with a starting index of $n = 0$ is given; and a complete and correct recursive definition is given. A *complete* recursive definition consists of a recurrence relation that correctly relates the current term to the previous term, *and* a correct initial condition. **Recursive definitions without the initial condition stated are not complete!**