Checkpoint information

Draft schedule of Checkpoints

Checkpoint	Date	LT's introduced
1	Sept 16	A.1, A.2
2	Sept 23	A.3, L.1
3	Sept 30	L.2, L.3, L.4
4	Oct 14	L.5, SF.1, SF.2
5	Oct 28	SF.3, SF.4, C.1, C.2
6	Nov 9	C.3, C.4
7	Nov 23	SR.1, SR.2, SR.3, SR.4
8	Dec 1	*None, just old LT's*
9	Dec 7	P.1, P.2
10	Dec 10	*None, just old LT's*
11 (Final exam)	Dec 14	*None, just old LT's*

Checkpoint Tasks and Grading rubrics for each Learning Target

- A.1: I can represent an integer in base 2, 8, 10, and 16.
    - Given: 3-5 integers in one of these four bases
    - To do: Convert each integer to 1-2 of the other bases
    - Check if: All work/explanation is clear, and all answers are correct.
- A.2 **(Core)**: I can add, subtract, multiply, and divide two integers written in binary.
    - Given: 3-4 pairs of binary integers, 8 bits or less
    - To do: Add, subtract, multiply, or divide each pair
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- A.3: I can compute `a % b` given integers `a` and `b` and perform arithmetic mod n.

    - Given: 3-4 pairs of integers a and b
    - To do: Compute `a % b` and, given a third integer n, compute sums, products, and powers mod n.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- L.1: I can use propositional variables and logical connectives to represent statements; and interpret symbolic logical statements in plain language.

    - Given: 2-4 expressions involving propositional variables and connectives, along with English meanings for each variable
    - To do: Give a correct and clear English interpretation of each expression.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- L.2 **(Core)**: I can write the negation, converse, and contrapositive of a conditional statement and use DeMorgan's Laws to simplify symbolic logical expressions.

    - Given: 2-3 conditional statements (either in symbolic or plain language form) and 1-3 logical expressions involving "and" and "or"
    - To do: State the negation, converse, and contrapositive of each conditional statement; use DeMorgan's Laws to simplify the and/or expressions
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- L.3: I can determine whether a quantified statement is true, false, or underdetermined, and state its negation.

    - Given: 2-3 quantified statements in symbolic or plain language form
    - To do: State whether each quantified statement is true, false, or indeterminate, and state the negation.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- L.4 **(Core)**: I can write the truth table for a logical statement.

    - Given: 2-3 logical statements with 2-3 variables in each
    - To do: Construct the truth table for each statement.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- L.5: I can determine if a statement is a tautology and whether two statements are logically equivalent.

    - Given: 1-2 standalone statements, and 1-2 pairs of other statements
    - To do: Use a truth table to determine if each standalone statement is a tautology, and whether each pair of statements is logically equivalent
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake

due to simple error.

- SF.1 **(Core)**: I can represent a set in roster notation and set-builder notation; determine if an object is an element of a set; and determine set relationships (equality, subset).
    - Given: 2-4 sets in either roster or set-builder notation or in plain language; 2-3 sets and objects; and 2-3 pairs of sets
    - To do: Rewrite each set in either set-builder or roster notation; determine whether each object is an element of its set or not; determine whether the sets in a pair are equal or if one is a subset of the other.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- SF.2: I can perform operations on sets (intersection, union, complement, Cartesian product) and determine the cardinality of a set.
    - Given: 2-4 pairs of sets
    - To do: Write out the intersection, union, and Cartesian product of at least one of the pairs; write the complement of several sets in the pairs; state the cardinality of several sets in the pairs.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- SF.3 **(Core)**: I can determine whether or not a given relation is a function, determine the domain and codomain of a function, and find the image and preimage of a point using a function.
    - Given: 3-4 relations on sets (some finite, some not) along with some points from each set
    - To do: Determine whether each relation is a function; if not, explain why; if so, state the domain and codomain and find the image and preimage of some of the points
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- SF.4: I can determine whether a function is injective, surjective, or bijective.
    - Given: 3-4 functions between sets (some finite, some not)
    - To do: Determine whether each function is injective, surjective, bijective, or none of these.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- C.1 **(Core)**: I can use the additive and multiplicative principles and the Principle of Inclusion and Exclusion to formulate and solve counting problems.
    - Given: A basic counting situation involving a combination of these principles
    - To do: Solve the counting problem.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- C.2: I can calculate a binomial coefficient and correctly apply the binomial coefficient to formulate

and solve counting problems.

  - Given: 2-3 “raw” binomial coefficients, and counting problems that use binomial coefficients
  - To do: Compute the binomial coefficients and solve the counting problems
  - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- C.3 **(Core)**: I can compute combinations and permutations and apply these to formulate and solve counting problems.

  - Given: 3-4 raw combinations and permutations; and counting problems that use combinations and permutations
  - To do: Compute the combinations and permutations, and solve the counting problems
  - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- C.4: I can use the “Stars and Bars” technique to formulate and solve counting problems.

  - Given: 1-2 counting situations involving distributing distinct items among a group
  - To do: Solve the counting problem
  - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- SR.1 **(Core)**: I can generate several values in a sequence defined using a closed-form expression or using recursion.

  - Given: 2-4 closed-form expressions or recursive definitions of sequences
  - To do: Generate the first 3-8 elements of the sequence
  - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- SR.2: I can use sigma notation to rewrite a sum and determine the sum of an expression given in sigma notation.

  - Given: 2-4 sums given in sigma notation or as sums
  - To do: Rewrite the sums in sigma notation, and find the value of each of the sums
  - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- SR.3 **(Core)**: I can find closed-form and recursive expressions for arithmetic and geometric sequences and find their sums.

  - Given: 2-3 sequences (of integers, strings, visual patterns, etc.)
  - To do: Generate recursive expressions for the sequences and (for numerical sequences) closed-form expressions
  - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.

- SR.4: I can use iteration and characteristic roots to solve a recurrence relation.
    - Given: 1-2 second-order linear recurrence relations
    - To do: Use either iteration or characteristic roots to solve each relation
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- P.1: I can analyze and write a combinatorial proof of a combinatorial identity.
    - Given: A combinatorial proof of an identity (for example, $\binom{n}{k} = \binom{n}{n-k}$)
    - To do: Perform a critical analysis of the proof
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.
- P.2 **(Core)**: Given a statement to be proven by (weak) induction, I can state and prove the base case, state the inductive hypothesis, and outline the proof.
    - Given: 1-3 mathematical conjectures that are to be proven by induction
    - To do: State and prove the base case and inductive hypotheses for each potential proof, then outline how the rest of the proof would go.
    - Check if: All work/explanation is clear, and all answers are correct; no more than one mistake due to simple error.