MTH 225 Learning Targets

Mastery of Learning Targets in groups A, L, SF, C, SR, and P are assessed using Checkpoints and the alternatives to Checkpoints described earlier. Learning Targets in group CS are *not* assessed by Checkpoints but rather by a portfolio as part of the Final Exam.

Group A: Represent integers using different number bases, and perform integer arithmetic using different bases and modular arithmetic.

- A.1: I can represent an integer in base 2, 8, 10, and 16.
- A.2 **(Core)**: I can add, subtract, multiply, and divide two integers written in binary.
- A.3: I can compute `a % b` given integers `a` and `b` and perform arithmetic mod n.

Group L: Formulate, manipulate, and determine the truth of logical expressions using symbolic logic.

- L.1: I can use propositional variables and logical connectives to represent statements; and interpret symbolic logical statements in plain language.
- 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.
- L.3: I can determine whether a quantified statement is true, false, or underdetermined, and state its negation.
- L.4 **(Core)**: I can write the truth table for a logical statement.
- L.5: I can determine if a statement is a tautology and whether two statements are logically equivalent.

Group SF: Formulate and solve computational problems using sets and functions.

- 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).
- SF.2: I can perform operations on sets (intersection, union, complement, Cartesian product) and determine the cardinality of a set.
- 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.
- SF.4: I can determine whether a function is injective, surjective, or bijective.

Group C: Formulate and solve complex counting problems using computational thinking and the tools of combinatorics.

- C.1 **(Core)**: I can use the additive and multiplicative principles and the Principle of Inclusion and Exclusion to formulate and solve counting problems.
- C.2: I can calculate a binomial coefficient and correctly apply the binomial coefficient to formulate and solve counting problems.
- C.3 **(Core)**: I can compute combinations and permutations and apply these to formulate and solve counting problems.
- C.4: I can use the “Stars and Bars” technique to formulate and solve counting problems.

Group SR: Evaluate numerical and other sequences using recursion, and solve simple recurrence relations.

- SR.1 **(Core)**: I can generate several values in a sequence defined using a closed-form expression or using recursion.
- SR.2: I can use sigma notation to rewrite a sum and determine the sum of an expression given in sigma notation.
- SR.3 **(Core)**: I can find closed-form and recursive expressions for arithmetic and geometric sequences and find their sums.
- SR.4: I can use iteration and characteristic roots to solve a recurrence relation.

Group P: Write clear, correct, and convincing arguments to explain the correctness of a solution using combinatorial proof and mathematical induction.

- P.1: I can analyze and write a combinatorial proof of a combinatorial identity.
- 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.

Group CS: Demonstrate problem solving, communication,and learning skills appropriate for computer science.

- CS.1: I can explain the reasoning behind solutions to computational problems clearly to an appropriate audience.
- CS.2: I can apply computer programming and computational thinking to frame and solve mathematical and computational problems.
- CS.3: I can self-assess my work and apply feedback from others to make improvements in my work.