Homework 1: Introduction to Proofs

Due: September 17, 2024 at 2:30p.m.

This homework must be typed in LaTeX and submitted via Gradescope.

Please ensure that your solutions are complete, concise, and communicated clearly. Use full sentences and plan your presentation before your write. Except where indicated, consider every problem as asking for a proof.

Problem 1. Let $f : A \to B$ and $g : B \to C$ be functions. Prove or disprove the following statement:

If $g \circ f$ is bijective, then f is injective and g is surjective.

Solution. □

Problem 2. Prove or disprove the following statement:

The sum of a rational number and an irrational number is irrational.

Solution. □

Problem 3. Prove or disprove the following statement:

The square for every odd number can be expressed as $8k + 1$ for some integer k.

Solution. □

Problem 4. You are given a rectangular chocolate bar with $m \times n$ squares of chocolate, and our task is to divide it into mn individual squares. You are only allowed to split one piece of chocolate at a time using a vertical or a horizontal break. For example, suppose that the chocolate bar is 2×2. The first split makes two pieces, both 2×1. Each of these pieces requires one more split to form single squares. This gives a total of three splits. Use an induction argument to prove the correctness of the following statement:

$mn - 1$ splits are sufficient to divide a rectangular chocolate bar with $m \times n$ squares into individual squares.

Solution. □

Problem 5. Consider the algorithm given as pseudocode below:

1. What is the output of the algorithm? Provide an informal but precise description.
2. Prove the correctness of the algorithm.
3. Analyze the running time of the algorithm.

Algorithm 1 ?-?

```
Input: An n-vertex graph represented by an adjacency matrix D, where D[i][j] is
the non-negative weight of the edge from vertex i to vertex j, for 0 ≤ i,j < n.
If there is no edge connecting i and j, D[i][j] = ∞.
Output: ?
for i ← 0 to n − 1 do                                    ▷ Initialization solution
   for j ← 0 to n − 1 do
      S[i][j] ← D[i][j]
   end for
end for
for k ← 0 to n − 1 do
   for i ← 0 to n − 1 do
      for j ← 0 to n − 1 do
         S[i][j] ← min{S[i][j], S[i][k] + S[k][j]}
      end for
   end for
end for
```