Solution 1: Introduction to Proofs

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. This statement is $\boxed{\text{true}}$. If you want a non-trivial example of two functions satisfying f and g (at least one that doesn't make f and g both bijective), let $A = C = \mathbb{R}$ and $B = \mathbb{R}^2$. Then with f and g defined as follows we have that,

$$f(x) := (x, 0)$$
$$g(x, y) := x$$

$g \circ f$ is the identity map. However, showing this example isn't enough to show that this is *always* the case, but it does provide nice intuition!

Proving f is injective. Suppose, for the sake of contradiction that f is not injective. Then, there exists two values $x, y \in A$ such that $x \neq y$ but $f(x) = f(y)$. Therefore, we have that

$$(g \circ f)(x) = g(f(x)) = g(f(y)) = (g \circ f)(y)$$

Therefore, $g \circ f$ is not injective, which contradicts the assumption that the map is bijective. Thus, f must be injective.

Proving g is surjective. Suppose, for the sake of contradiction that g is not surjective. Then, there exists a value $c \in C$ such that for all $b \in B$, $g(b) \neq c$. Therefore, there exists no $a \in A$ such that $(g \circ f)(a) = c$, which contradicts the assumption that the map is bijective. Thus g must be surjective. □

Problem 2. Prove or disprove the following statement:

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

Solution. The statement is $\boxed{\text{true}}$. Let a be a rational number and b be an irrational number. For the sake of contradiction, suppose that $a+b$ is rational. Because $a, a+b$ are rational, they can be written as $a = m/n$ and $a+b = x/y$ for integers m, n, x, y (such that $n, y \neq 0$). Then,

$$\frac{m}{n} + b = \frac{x}{y}$$

$$b = \frac{x}{y} - \frac{m}{n}$$

$$b = \frac{xn - ym}{yn}$$

Because m, n, x, y are integers, $xn - ym$ and yn are integers. Furthermore, yn is non-zero due to the assumption that $n, y \neq 0$, so by the definition of rational numbers, b is rational. This contradicts the fact that b is irrational, which means our original assumption that $a+b$ is rational is false. □

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. The statement is $\boxed{\text{true}}$. Let n be odd: this means that it can be expressed $n = 2m + 1$, for some integer m. Squaring,

$$n^2 = (2m + 1)^2 = 4m^2 + 4m + 1.$$

Factoring out the 4, we can rewrite this as

$$n^2 = 4(m^2 + m) + 1.$$

Consider $m^2 + m$: this can be factored as $m(m+1)$, which is a product of two consecutive integers. Because they are consecutive, this is a product of an even and odd integer, which is always even. Therefore, we can find some integer k so that $m(m + 1) = 2k$. Plugging this back in,

$$n^2 = 4(m^2 + m) + 1$$

$$n^2 = 4(2k) + 1$$

$$n^2 = 8k + 1$$

which is exactly the statement we want to prove. □

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. Strong induct on the size of the chocolate bar, mn. Let $P(m, n)$ be the proposition that a chocolate bar with $m \times n$ squares requires at most $mn - 1$ splits.

Base Case: The base case is $mn = 1$, or $m = n = 1$. This holds trivially because we already start with an individaul square, so $0 = 1 \cdot 1 - 1$ splits are needed.

Inductive Hypothesis: Suppose that P holds on all m, n satisfying $mn \leq k$.

Induction Step: We want to show that P holds for all m, n satisfying $mn = k+1$. Since $k+1 > 1$, one of m, n must be greater than 1: without loss of generality, take $m > 1$. Then, split along any horizontal break to split the bar into a $p \times n$ and $q \times n$ piece, where $p + q = m$ and $p, q > 1$. Now, we have two chocolate bars with size $pn, qn < mn = k + 1$. This is the same as $pn, qn \leq k$, so we can apply the inductive hypothesis and conclude that it is possible to split the $p \times n$ and $q \times n$ chocolate bar in at most $pn - 1$ and $qn - 1$ moves, respectively. We now have a way to split the $m \times n$ bar in at most

$$1 + (pn - 1) + (qn - 1) = (p + q)n - 1 = mn - 1$$

splits. Thus, we have shown $P(m, n)$ is true for all m, n satisfying $m \times n = k + 1$, which completes the induction.

□

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
```

Solution. 1. The algorithm returns a matrix S, where $S[i][j]$ represents the weight of the shortest path between i and j in the graph represented by D (where we view weights here as distance).

Remark: This algorithm is known as the **Floyd Warshall** algorithm.

2. Here, we choose to induct on k in the outermost for-loop in the main logic. The claim is that after the mth iteration, $S[i][j]$ will represent the length of the shortest path from i to j that uses only the vertices $\{0, \cdots, m-1\}$ as intermediate vertices. Formally, if our path is $\{p_0, p_1, \cdots, p_d\}$, where $p_0 = i$ and $p_d = j$, then $p_1, \cdots, p_{d-1} \in \{0, \cdots m-1\}$.

   For the base case of $k = 1$, we wish to show that $S[i][j]$ is the shortest path from i to j passing through only 0 as an intermediate vertex. This is true: if the path uses 0 as an intermediate vertex, it must look like $i \to 0 \to j$, so this is represented by $S[i][0] + S[0][j]$. Then, we take the minimum with the current value of $S[i][j]$, which is the weight of the edge from i to j, meaning $S[i][j]$ now holds the length of the shortest path from i to j with all intermediate vertices in the set $\{0\}$.

   Now suppose the hypothesis holds for the mth iteration. We wish to show that it holds after the $m+1$th iteration. If we don't pass through vertex m at all, then the shortest path is precisely $S[i][j]$, since now the intermediate vertex set is limited to $\{0, \cdots, m-1\}$.

Suppose now that the path passes through m. It is clear that we don't want any cycles in this path, so the path from i to m and the path from m to j must not pass through m again. However, this means that the intermediate vertices from $i \to m$ and $m \to j$ are precisely in the set $\{0, \cdots, m-1\}$! This means we can use the inductive hypothesis and conclude the shortest length path from $i \to m$ is $S[i][m]$ and from $m \to j$ is $S[m][j]$. Thus, in the case that we pass through m, the shortest distance is $S[i][m] + S[m][j]$. Therefore, setting $S[i][j] = \min\{S[i][j], S[i][m] + S[m][j]\}$ properly represents the shortest path from i to j using only intermediate vertices from $\{0, \cdots, m\}$, completing the induction.

To finish, we note that after all the iterations of the outermost for-loop, $S[i][j]$ represents the shortest path between i and j using intermediate vertices in $\{0, \cdots, n-1\}$. This is now the entire vertex set in our graph, so it is the same as the shortest path between i and j in the graph, as claimed in (a).

3. The running time of the algorithm is $O(n^3)$: the initialization of S requires two nested for loops, which is $O(n^2)$, and the main logic of the algorithm is a triple nested for loop with an $O(1)$ operation in each loop, which means it is in $O(n^3)$. Thus, the overall runtime is $O(n^2) + O(n^3) \in O(n^3)$.

□