CSC236 Problem Set 1

Xuanqi Wei

September 17, 2023

Contents

1 Question 1

(a) According to the definition of P:

$$\forall g_1 \in G_1,\ \exists t_1 \in T_1,\ t_1\ tiles\ g_1 \implies \forall g_2 \in G_2,\ \exists t_2 \in T_2,\ t_2\ tiles\ g_2$$

(b) Firstly, assume

$$\forall g_1 \in G_1,\ \exists t_1 \in T_1,\ t_1\ tiles\ g, \text{ which is the antecedent.}$$

Secondly, I will do the consequent part, which:

$$Let\ g_2\ be\ an\ arbitrary\ element\ from\ G_2$$

Then, I want to prove that

$$\exists t_2 \in T_2,\ t_2\ tiles\ g_2$$

by selecting a satisfying element t_2 from T_2 and prove the element t_2 satisfies t_2 *tiles* g_2.

(c) The diagram above illustrates one instance of G_2 grids, which being tiled by triominoes.

Firstly, we already know that for P(1), the statement $\forall g_1 \in G_1,\ \exists t_1 \in T_1,\ t_1\ tiles\ g$ is true which is the antecedent of this direct proof.

Secondly, the above diagram is an element of the set of all $2^2 \times 2^2$ grid with one square removed, which is an element of G_2. By visulising those colorful triominoes, we see a combination triominoes, t_2, which is an element of the set of all tilings of elements of G_2 using triominoes, belonging to T_2, exists and tiles g_2.

Therefore, the diagram above illustrates an instance of that direct proof.

(d) Given the statement to prove: $\forall n \in \mathbb{N},\ P(n)$, which for each natural n you can tile any $2^n \times 2^n$ grid with one cell missing using only triominoes.

Proof: We prove this by Simple Induction on n.

Base Case: Let $n = 1$.

Since G_1 is the set of all $2^1 \times 2^1$ grids with one cell removed, which by definition is a single triominoe.

Therefore, $\forall g_1 \in G_1,\ \exists t_1 \in T_1,\ t_1\ tiles\ g_1$ is true, which P(1) is true.

Induction Step: Let $k \in \mathbb{N}$.

Induction Hypothesis: Assume that $P(k)$ is true.

By Induction Hypothesis, we know that $P(k)$ is true, which $\forall g_k \in G_k,\ \exists t_k \in T_k,\ t_k\ tiles\ g_k$ is true. I will take 3 different g_ks, the first with right button corner square missing, the second with right top corner square missing, and the third with left top corner square missing. I will make the missing corners in these 3 g_ks face inwards and add a triomino which will result in getting a 'L' shape. The remaining $\frac{1}{4}$ place is missing a cell to form a g_{k+1}, which can actually be an arbitraty element from G_k. By Induction Hypothesis, since $\forall g_k \in G_k,\ \exists t_k \in T_k,\ t_k\ tiles\ g_k$ is true, the remaining G_k place can be covered by trimonoes, proving the $P(k+1)$ is true.

Therefore, we've proved $\forall n \in \mathbb{N},\ P(n)$ is true.

■

Question 2

```python
from typing import Any


def q_2(n: int, x: Any) -> Any:
    """Implement a Python function with parameters x and n that (ignoring f

    Precondition:
    1. x represents a non-zero real number.
    2. n is a natural number
    """
    # Since c_1 is used in both n = 1 and recursion, I will put it at the f
    c_1 = x + 1 / x

    if n == 1:
        # From the definition of c_n, when n is 1, return the corresponding
        return c_1
    elif n == 2:
        # From the definition of c_n, when n is 2, return the corresponding
        c_2 = x * x + (1 / x) * (1 / x)
        return c_2
    else:
        """This is the recursion part. According to the discovery from hint
        general function for c_n.
        """
        # Aim at returning the recursive value of c for n minus 1 after rea
        c_minus1 = q_2(n-1, x)
        # Aim at returning the recursive value of c for n minus 2. Since we
        # an odd number, we need to add both n equals to 1 and n equals to
        c_minus2 = q_2(n-2, x)
        # Calculate the c_n based on the discovery.
        c_n = c_1 * c_minus1 - c_minus2
        return c_n


if __name__ == '__main__':
    test_n = 5
    test_x = 5
    print(pow(test_x, test_n) + pow((1 / test_x), test_n))
    print(q_2(test_n, test_x))
```

(a)
```python
print("Hellp, World!")
```