Solutions To HW6

Zixuan Chen
X313572
zixuanchen@ucsb.edu

July 2024

Solution To Question 1

Write the 6 terms of the sequence that is described by each of the recurrence relations below:
(a) $f_1 = 0, f_2 = 2,$ *and* $f_n = 5f_{n-1} - 2f_{n-2}$ *for* $n \geq 3$

- $f_1 = 0$
- $f_2 = 2$
- $f_3 = 5f_2 - 2f_1 = 10 - 0 = 10$
- $f_4 = 5f_3 - 2f_2 = 5 * 10 - 2 * 2 = 46$
- $f_5 = 5 * 46 - 2 * 10 = 210$
- $f_6 = 5 * 210 - 2 * 46 = 958$

(b) $g_1 = 2$ *and* $g_2 = 1$. The rest of the terms are given by the formula $g_n = ng_{n-1} + g_{n-2}$

- $g_1 = 2$
- $g_2 = 1$
- $g_3 = 3 * 1 + 2 = 5$
- $g_4 = 4 * 5 + 1 = 21$
- $g_5 = 5 * 21 + 5 = 110$
- $g_6 = 6 * 110 + 21 = 681$

Solution To Question 2

$$a_0 = a_1 = 2$$
$$a_n = a_{n-1}^2 a_{n-2}, \quad n \geq 2$$

Prove using strong induction:

$$\forall n \in \mathbb{Z}^{\geq 0} \left(a_n \leq 2^{3^n} \right)$$

Base Case:

For $n = 0, n = 1$:

$$a_0 = 2 \leq 2^{3^0} = 2^1 = 2$$
$$a_1 = 2 \leq 2^{3^1} = 2^3 = 8$$

Inductive Steps:

Assume for $k \leq n$, we have $a_k \leq 2^{3^k}$, we need to show that $a_{n+1} \leq 2^{3^{n+1}}$

From definition, we know:

$$a_{n+1} = a_n^2 a_{n-1}$$

From the assumption we have:

$$a_n^2 a_{n-1} \leq 2^{3^n} \cdot 2^{3^{n-1}} = 2^{3^n + 3^{n-1}}$$

Now we have

$$a_{n+1} \leq 2^{3^n + 3^{n-1}}$$

So if we prove that $2^{3^n + 3^{n-1}} \leq 2^{3^{n+1}}$, we can prove the conclusion. As function 2^x is an increasing function, we only need to prove:

$$3^n + 3^{n-1} \leq 3^{n+1}$$
$$3^n + \frac{1}{3} 3^n \leq 3^n + 2 \cdot 3^n$$

Obviously, the inequality holds true. Now, combining the two inequalities:

$$2^{3^n + 3^{n-1}} \leq 2^{3^{n+1}}$$

$$a_{n+1} \leq 2^{3^n + 3^{n-1}}$$

We have:

$$a_{n+1} \leq 2^{3^{n+1}}$$

Therefore, the conclusion is proved.

Solution To Question 3

Write a recursive algorithm to compute the maximum of a sequence of numbers. Then, use induction to prove that your algorithm outputs the correct value for every non-empty input sequence.

Algorithm: Compute the maximum of a sequence of numbers

```
procedure max(seq)
    if length(seq) == 1 then
        return seq[0]

    now_max = seq[0]
    sub_max = max(seq[1:])
    if now_max > sub_max then
        return now_max
    else
        return sub_max
```

Proof:

Base Case: For a sequence of length 1, the only element is trivially the largest element.

Inductive Step: Assume that for a sequence seq of length n, the algorithm correctly finds the maximum element m. That is, $\max(\text{seq}) = m$.

Now consider a sequence of length $n+1$, $\text{seq'} = (x, \text{seq})$.

According to the algorithm, the maximum of seq' is computed as:

$$\max(\text{seq'}) = \max(x, \max(\text{seq}))$$

By the induction hypothesis, we know $\max(\text{seq}) = m$.

Then, there are two cases to consider:

1. If $x < m$, then $\max(x, m) = m$, so $\max(\text{seq'}) = m$, which means m is the maximum element in seq'.

2. If $x \geq m$, then $\max(x, m) = x$, so $\max(\text{seq'}) = x$, which means x is the maximum element in seq'.

In both cases, the algorithm correctly finds the maximum element in seq'.

Thus, by induction, the algorithm correctly finds the maximum element in any non-empty input sequence.

Solution To Question 4

$$P(n) : \sum_{j=1}^{n} j^2 = n(n+1)(2n+1)/6$$

(a) Verify $P(3)$, express $P(k), P(k+1)$

$$\sum_{j=1}^{3} j^2 = 1 + 4 + 9 = 14$$

$$\frac{3 * 4 * 7}{6} = 14$$

Therefore, $P(3)$ is true.

$$P(k) : \sum_{j=1}^{k} j^2 = k(k+1)(2k+1)/6$$

$$P(k+1) : \sum_{j=1}^{k+1} j^2 = (k+1)(k+2)(2(k+1)+1)/6$$

(b) What is the basis step for an inductive proof of $\forall n \in \mathbb{Z}^+(P(n))$

$$P(1) : \sum_{j=1}^{1} j^2 = 1^2 = 1 = \frac{1(1+1)(2+1)}{6}$$

(c) What would be the inductive hypothesis? What must be proven in the inductive step?

Assume that $P(k)$ is true for some $k \geq 1$:

$$\sum_{j=1}^{k} j^2 = k(k+1)(2k+1)/6$$

And we need to prove $P(k+1)$ is true:

$$\sum_{j=1}^{k+1} j^2 = (k+1)(k+2)(2(k+1)+1)/6$$

(d) Complete inductive proof

Now we have:

$$\sum_{j=1}^{k} j^2 = k(k+1)(2k+1)/6$$

and

$$\sum_{j=1}^{k+1} j^2 = \sum_{j=1}^{k} j^2 + (k+1)^2$$

We need to prove that:

$$\sum_{j=1}^{k+1} j^2 = (k+1)(k+2)(2(k+1)+1)/6$$

Combining these equations:

$$\sum_{j=1}^{k+1} j^2 = k(k+1)(2k+1)/6 + (k+1)^2$$

Now we want to prove:

$$k(k+1)(2k+1)/6 + (k+1)^2 = (k+1)(k+2)(2(k+1)+1)/6$$

Multiply 6 to both sides:

$$k(k+1)(2k+1) + 6(k+1)^2 = (k+1)(k+2)(2k+3)$$

Dividing both side by $k+1$:

$$k(2k+1) + 6(k+1) = (k+2)(2k+3)$$

Expanding both sides:

$$2k^2 + k + 6k + 6 = 2k^2 + 3k + 4k + 6$$

$$2k^2 + 7k + 6 = 2k^2 + 7k + 6$$

Obviously, the equation holds true.

Therefore, the assertion is true.

Solution To Question 5

Prove $n_0 \in \mathbb{N}\ \forall n \in \mathbb{Z}^{\geq n_0}\ (n^2 < 2^n)$ with lemma: $n_0 \in \mathbb{N}\ \forall n \in \mathbb{Z}^{\geq n_0}\ (1 + 2n < n^2)$.

Proof:

First, we choose the witness n_0 to be 5.

Base Case: $n = 5$. Obviously, $n^2 = 25 < 2^n = 32$

Inductive Steps:

Assume that: for some $k \geq 1$, $k^2 < 2^k$.

We need to prove: $(k + 1)^2 < 2^{k+1}$.

Expand both sides:

$$k^2 + 2k + 1 < 2^k + 2^k$$

As $k^2 < 2^k$, we only need to prove:

$$2k + 1 < 2^k$$

Combining the lemma:

$$1 + 2k < k^2$$

and assumption

$$k^2 < 2^k$$

we have:

$$2k + 1 < 2^k$$

Therefore, the statement is proved by induction with witness $n_0 = 5$.

Solution To Question 6

Can the statement you proved above be used to prove or disprove the following statement?

$$n_0 \in \mathbb{N}\ \forall n \in \mathbb{Z}^{\geq n_0}\ (2^n < n^2)$$

In Question 5, we proved:

$$n_0 \in \mathbb{N}\ \forall n \in \mathbb{Z}^{\geq n_0}\ (n^2 < 2^n)$$

We denote the two statements as A and B respectively.

Assume that A is true. So we can choose a witness n_0 as a.

And we also choose a witness n_0 for B as b.

So now we have:

$$\forall n \geq a, 2^n < n^2$$
$$\forall n \geq b, n^2 < 2^n$$

Therefore, we can always choose a number $x = max\{a, b\}$ and we will have:

$$2^x < x^2 \wedge x^2 < 2^x$$

which is obviously false.

Therefore, the statement is false.

Solution To Question 7

Base Case: For $n = 0$,

$$\text{StringSet}(0) = \{\lambda\}$$

The only string of length 0 is the empty string.

Inductive Steps:

Assume that the algorithm correctly computes the set of binary strings of length k.

For $n = k + 1$:

$$\text{StringSet}(k + 1) = \{0x, 1x \mid x \in \text{StringSet}(k)\}$$

By the induction hypothesis,$\text{StringSet}(k)$ correctly computes all binary strings of length k. The concatenation of 0 and 1 to each of these strings correctly generates all binary strings of length $k + 1$.

Therefore, by induction, the algorithm correctly computes the set of all binary strings of length n.

Solution To Question 8

First, the function is well-defined because if we choose an arbitrary 4-bit 1/0 string $abcd$, it will be mapped to bcd, which is in the codomain because it's a 3-bit 1/0 string.

The function is **not one-to-one** because it will map both 0111 and 1111 to 111.

The function is **onto**.

Choose an arbitrary 3-bit 1/0 string x in the codomain. And we can find both $1x$ and $0x$ are in the domain and they're mapped to x.