Lab01-Proof

CS363-Computability Theory, Xiaofeng Gao, Spring 2016

∗ Please upload your assignment to TA's FTP. Contact nongeek.zv@gmail.com for any questions.

∗ Name:Wenhao Zhu StudentId: 5130309717 Email: weehowe.z@gmail.com

1. Prove that for any integer $n > 2$, there is a prime p satisfying $n < p < n!$. (Hint: consider a prime factor p of $n! - 1$ and use proof by contradiction)

   Proof. Assume that for any integer $n > 2$, there is no prime p satisfying $n < p < n!$. That means for any x satisfying $n < x < n!$ is not a prime.Then we consider a prime factor p of $n! - 1$, if $p <= n$, then p is also a prime factor of $n!$ according to the definition of factorial. Since $p|n!$ and $p|n! - 1$, we know $p|1$, there is a contradiction. So p must be $p > n$, then we find a prime p satisfying $n < p < n!$ which contradicts the assumption. Therefore, for any integer $n > 2$, there is a prime p satisfying $n < p < n!$. □

2. Use minimal counterexample principle to prove that: for every integer $n > 17$, there exist integers $i_n \geq 0$ and $j_n \geq 0$, such that $n = i_n \times 4 + j_n \times 7$.

   Proof. If $n = i_n \times 4 + j_n \times 7$ is not true for every integer $n > 17$, then there are values of n for which makes the equation false, and there must be a smallest such value, say $n = k$. Since $18 = 1 \times 4 + 2 \times 7$, $19 = 3 \times 4 + 1 \times 7$, $20 = 5 \times 4 + 0 \times 7$. we have $k \geq 21$. Since k is the smallest value, we know $k - 1$ satisfying the equation, which means there exist integers $i_{k-1} \geq 0$ and $j_{k-1} \geq 0$, such that $k - 1 = i_{k-1} \times 4 + j_{k-1} \times 7$. However, we have:

$$
\begin{aligned}
& k = i_{k-1} \times 4 + j_{k-1} \times 7 + 1 \\
\Leftrightarrow \quad & k = (i_{k-1} + 2) \times 4 + (j_{k-1} - 1) \times 7 \\
\Leftrightarrow \quad & k = (i_{k-1} - 5) \times 4 + (j_{k-1} + 1) \times 7
\end{aligned}
$$

   (1) If $j_{k-1} \geq 1$, denote $i_k = i_{k-1} + 2$, $j_k = j_{k-1} - 1$, then we know $i_k > i_{k-1} \geq 0$, $j_k = j_{k-1} - 1 \geq 0$. Thus $n = i_n \times 4 + j_n \times 7$ is still true, we have derived a contradiction.
   (2) If $j_{k-1} - 1 = 0$, then we know $i_{k-1} = (k-1)/4 \geq 5$, denote $i_k = i_{k-1} - 5$, $j_k = j_{k-1} + 1$. We can get $i_k = i_{k-1} - 5 \geq 0$ and $j_k = j_{k-1} + 1 \geq 0$. Thus $n = i_n \times 4 + j_n \times 7$ is still true, we have derived a contradiction.
   In conclusion, for every integer $n > 17$, there exist integers $i_n \geq 0$ and $j_n \geq 0$, such that $n = i_n \times 4 + j_n \times 7$.

   □

3. Suppose $a_0 = 1$, $a_1 = 2$, $a_2 = 3$, $a_k = a_{k-1} + a_{k-2} + a_{k-3}$ for $k \geq 3$. Use strong principle of mathematical induction to prove that $a_n \leq 2^n$ for all integers $n \geq 0$.

   Proof. Define $P(n)$ be the statement that $a_n \leq 2^n$. We will try to prove that $P(n)$ is true for every integer $n \geq 0$.
   Basis step. $P(0) = 1 \leq 2^0$, $P(1) = 2 \leq 2^1$, $P(2) = 3 \leq 2^2$, thus $P(0)$, $P(1)$, $P(2)$ are true.
   Induction hypothesis. For $k \geq 0$ and $0 \leq n \leq k$, $P(n)$ is true. (Strong Principle)
   Proof of induction step. Then we prove $P(k+1)$.

$$
\begin{aligned}
a_{k+1} &= a_k + a_{k-1} + a_{k-2} \\
&\leq 2^k + 2^{k-1} + 2^{k-2} \\
&\leq 7 \times 2^{k-2} \\
&\leq 2^{k+1}
\end{aligned}
$$

$P(k+1)$ is true, thus $a_n \leq 2^n$ for all integers $n \geq 0$. □

4. Consider the following loop, written in pseudocode:

```
while B do
  | S;
end
```

A condition P is called an invariant of the loop if whenever P and B are both true, and S is executed once, P is still true.

(a) Prove that if P is an invariant of the loop, and P is true before the first iteration of the loop, then if the loop eventually terminates (i.e., after some number of iterations, B is false), P is still true.

Proof. Suppose the loop totally runs n iterations, thus B is true in the first n iterations and change to false in the $n+1$ iteration. As P is an invariant of the loop, and it is true before the loop, thus P and B are both true at the beginning, therefore, after the iteration, P is still true. This situation remains for n iterations until B change to false. At the end of this iteration, P is true. Then in the next iteration, S won't be excuted, nothing has changed, so P remains to be true. □

(b) Suppose x and y are integer variables, and initally $x \geq 0$ and $y > 0$. Consider the following program fragment:

```
q = 0;
r = x;
while r ≥ y do
  | q = q + 1;
  | r = r − y;
end
```

By considering the condition $(r \geq 0) \wedge (x = q \times y + r)$, prove that when this loop terminates, the values of q and r will be the integer quotient and remainder, respectively, when x is divided by y; in other words, $x = q \times y + r$ and $0 \leq r < y$.

Proof. Define P is $x = q \times y + r$ and $0 \leq r < y$. Then we prove P is invariant.
Before the iteration, as $r = x$, $x = 0 \times y + x = q \times y + r$, P is true.
Whenever P and $r \geq y$ are true, during the iteration, $q^{'} = q + 1$, $r^{'} = r - y$, $x^{'} = x$, $y^{'} = y$. $q^{'}$, $r^{'}$ are the value of p, q after the iteration. We know: $x^{'} = x = q \times y + r = (q+1) \times y + r - y = q^{'} \times y^{'} + r^{'}$. Therefore, P is still true. So P is invariant. Thus, when this loop terminates, P is true, which means $x = q \times y + r$, and $r \geq y$ is false when r is smaller than y and r never minus a number larger than itself, so $0 \leq r < y$. □