1 Natural Induction on Inequality

Prove that if $n \in \mathbb{N}$ and $x > 0$, then $(1+x)^n \geq 1+nx$.

Solution:

- *Base Case:* When $n = 0$, the claim holds since $(1+x)^0 \geq 1+0x$.
- *Inductive Hypothesis:* Assume that $(1+x)^k \geq 1+kx$ for some value of $n = k$ where $k \in \mathbb{N}$.
- *Inductive Step:* For $n = k+1$, we can show the following:

$$\begin{aligned}(1+x)^{k+1} = (1+x)^k(1+x) &\geq (1+kx)(1+x) \\ &\geq 1+kx+x+kx^2 \\ &\geq 1+(k+1)x+kx^2 \geq 1+(k+1)x\end{aligned}$$

By induction, we have shown that $\forall n \in \mathbb{N}, (1+x)^n \geq 1+nx$.

2 Strengthen Induction

Show by induction that $\sum_{i=1}^{n} \frac{1}{i^3} \leq 2$.

Solution:

We will prove the stronger theorem that $\sum_{i=1}^{n} \frac{1}{i^3} \leq 2 - \frac{1}{n^2}$ by induction.

Base case: For $n = 1$, $1 \leq 2 - \frac{1}{(1)^2} = 1$.

Induction hypothesis: $\sum_{i=1}^{k} \frac{1}{i^3} \leq 2 - \frac{1}{k^2}$. $\sum_{i=1}^{k+1} \frac{1}{i^3} \leq (2 - \frac{1}{k^2}) + \frac{1}{(k+1)^3} = 2 - (\frac{1}{k^2} - \frac{1}{(k+1)^3})$ Now to complete the proof, we need to prove

$$\begin{aligned}\frac{1}{(k+1)^2} &\leq \frac{1}{k^2} - \frac{1}{(k+1)^3} \\ \frac{1}{(k+1)^2} &\leq \frac{1}{k^2} - \frac{1}{(k+1)^3} \\ \frac{k+2}{(k+1)^3} &\leq \frac{1}{k^2} \\ \frac{k^3+2k^2}{(k^2)(k+1)^3} &\leq \frac{(k+1)^3}{(k^2)(k+1)^3} \\ 0 &\leq \frac{k^2+3k+1}{(k^2)(k+1)^3}\end{aligned} \tag{1}$$

Thus the inequality holds, and the statement follows.

3 Binary Numbers

Prove that every positive integer n can be written in binary. In other words, prove that we can write

$$n = c_k \cdot 2^k + c_{k-1} \cdot 2^{k-1} + \cdots + c_1 \cdot 2^1 + c_0 \cdot 2^0,$$

where $k \in \mathbb{N}$ and $c_i \in \{0,1\}$ for all $i \leq k$.

Solution:

Prove by strong induction on n.

The key insight here is that if n is divisible by 2, then it is easy to get a bit string representation of $(n+1)$ from that of n. However, if n is not divisible by 2, then $(n+1)$ will be, and its binary representation will be more easily derived from that of $(n+1)/2$. More formally:

- Base Case: $n = 1$ can be written as 1×2^0.
- Inductive Step: Assume that the statement is true for all $1 \leq m \leq n$, where n is arbitrary. Now, we need to consider $n+1$. If $n+1$ is divisible by 2, then we can apply our inductive hypothesis to $(n+1)/2$ and use its representation to express $n+1$ in the desired form.

$$(n+1)/2 = c_k \cdot 2^k + c_{k-1} \cdot 2^{k-1} + \cdots + c_1 \cdot 2^1 + c_0 \cdot 2^0$$
$$n+1 = 2 \cdot (n+1)/2 = c_k \cdot 2^{k+1} + c_{k-1} \cdot 2^k + \cdots + c_1 \cdot 2^2 + c_0 \cdot 2^1 + 0 \cdot 2^0.$$

  Otherwise, n must be divisible by 2 and thus have $c_0 = 0$. We can obtain the representation of $n+1$ from n as follows:

$$n = c_k \cdot 2^k + c_{k-1} \cdot 2^{k-1} + \cdots + c_1 \cdot 2^1 + 0 \cdot 2^0$$
$$n+1 = c_k \cdot 2^k + c_{k-1} \cdot 2^{k-1} + \cdots + c_1 \cdot 2^1 + 1 \cdot 2^0$$

  Therefore, the statement is true.

Here is another alternate solution emulating the algorithm of converting a decimal number to a binary number.

- Base Case: $n = 1$ can be written as 1×2^0.
- Inductive Step: Assume that the statement is true for all $1 \leq m \leq n$, for arbitrary n. We show that the statement holds for $n+1$. Let 2^m be the largest power of 2 such that $n+1 \geq 2^m$. Thus, $n+1 < 2^{m+1}$. We examine the number $(n+1) - 2^m$. Since $(n+1) - 2^m < n+1$, the inductive hypothesis holds, so we have a binary representation for $(n+1) - 2^m$. Also, since $n+1 < 2^{m+1}$, $(n+1) - 2^m < 2^m$, so the largest power of 2 in the representation of $(n+1) - 2^m$ is 2^{m-1}. Thus, by the inductive hypothesis,

$$(n+1) - 2^m = c_{m-1} \cdot 2^{m-1} + c_{m-2} \cdot 2^{m-2} + \cdots + c_1 \cdot 2^1 + c_0 \cdot 2^0,$$

and adding 2^m to both sides gives

$$n+1 = 2^m + c_{m-1}\cdot 2^{m-1} + c_{m-2}\cdot 2^{m-2} + \cdots + c_1\cdot 2^1 + c_0\cdot 2^0,$$

which is a binary representation for $n+1$. Thus, the induction is complete.

Another intuition is that if x has a binary representation, $2x$ and $2x+1$ do as well: shift the bits and possibly place 1 in the last bit. The above induction could then have proceeded from n and used the binary representation of $\lfloor n/2 \rfloor$, shifting and possibly setting the first bit depending on whether n is odd or even.

Note: In proofs using simple induction, we only use $P(n)$ in order to prove $P(n+1)$. Simple induction gets stuck here because in order to prove $P(n+1)$ in the inductive step, we need to assume more than just $P(n)$. This is because it is not immediately clear how to get a representation for $P(n+1)$ using just $P(n)$, particularly in the case that $n+1$ is divisible by 2. As a result, we assume the statement to be true for all of $1,2,\ldots,n$ in order to prove it for $P(n+1)$.