Let $k, c \in \mathbb{N}$. We define $f \colon \mathbb{N} \to \mathbb{N}$ as

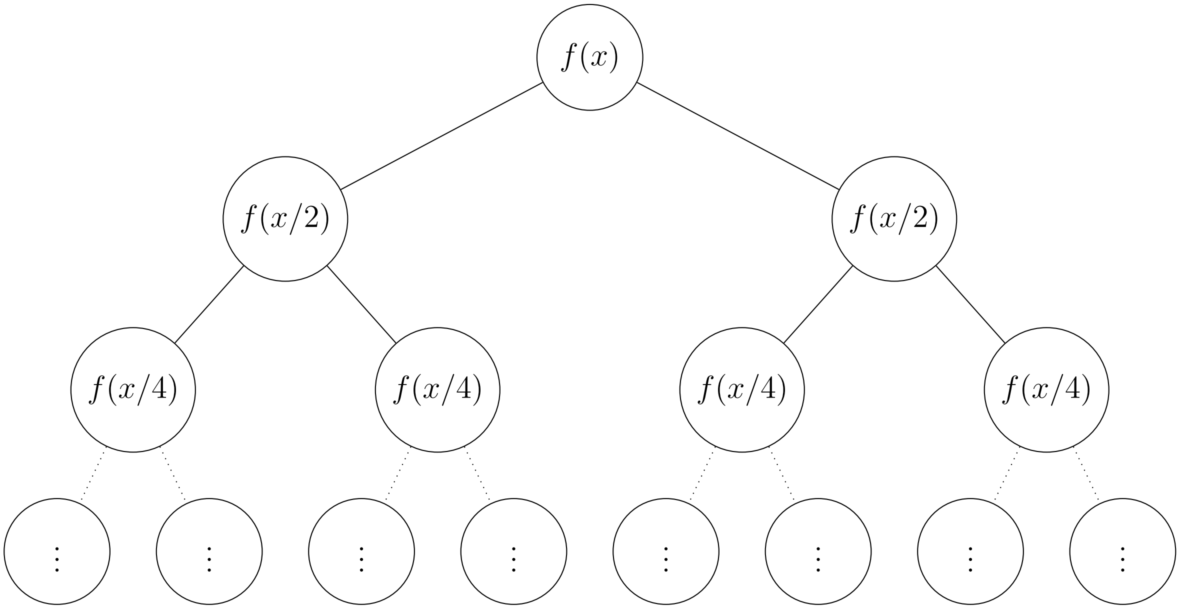
$$f(x) = \begin{cases} c, & \text{if } x < k \\ k \cdot f(x/k) + c, & \text{otherwise} \end{cases}$$

Then, f is $O(n)$.

Proof. We see that the constant at the end of the function does not impact the asymptotic behavior. This is because even if we define $f(x) = f(x-1) + c$ the most amount of c's we add is x so $g(x) = x \cdot c$ is in fact $O(x)$ linear. So now we are left with

$$f(x) = \begin{cases} 1, & \text{if } x < k \\ k \cdot f(x/k), & \text{otherwise} \end{cases}$$

for all the small cases $x < k$. For $k = 2$ the recursion will look as the illustration below.



We see that this is the equivalent problem to calculating the time complexity of building a full k-ary tree where there are x leaf nodes. Since for each leaf node, the path from the root is at most $\log_k(x)$ so $f(x)$ is $O(x \log_k(x))$ directly. But we can bound this even tighter, since there are much less than $O(\log_k(x))$ nodes in the tree. We prove the following first.

Lemma 0.1. $\sum_{i=0}^{m} a^i < a^{m+1}$ *for* $a, m \geq 2$

Proof. **Check for** $m = 2$**:**
$\sum a^i = a^0 + a + a^2 = 1 + a + a^2 < a^3 \iff \frac{1}{a^2} + \frac{1}{a} + 1 < a$
$\frac{1}{a^2} + \frac{1}{a} + 1 \leq \frac{1}{4} + \frac{1}{2} + 1 = \frac{7}{4} < 2 \leq a$
Assume $\sum_{i=0}^{m} a^i < a^{m+1}$ for some m.
Then $\sum_{i=0}^{m} a^i + a^{m+1} < a^{m+1} + a^{m+1} \iff \sum_{i=0}^{m+1} a^i < 2a^{m+1} \leq a^{m+2}$
So by induction we get that the hypothesis was correct. □

Now notice that in each level of the tree we have k^i nodes in total. So there are in total

$$\sum_{i=0}^{\log_k(x)} k^i$$

in total. And from the small lemma we proved we know that

$$\sum_{i=0}^{\log_k(x)} k^i < k^{\log_k(x)+1} = kx$$

Therefore there are in total kx nodes so $f(x)$ is $O(x)$. $\square$