

CS320 Programming Languages

Solution Set 2

Nenad Dedić
Computer Science Department
Boston University

Exercise 1: We will prove the statement using mathematical induction.

Let $P(n) : \mathcal{N} \rightarrow \{T, F\}$ be a predicate defined as follows (\mathcal{N} here denotes the set of all non-zero natural numbers): $P(n) = T$ if and only if all Braun trees with at most n nodes are unique (that is, for any $k \leq n$ there is exactly one Braun tree with k nodes).

If we prove that if $P(n)$ is true for all natural $n \geq n_0$, then for all natural $n \geq 1$ there is exactly one Braun tree with n nodes (n_0 will be the base of induction; in our case it will be 2).

Now, here is an inductive proof that P is true for all non-zero natural numbers:

- **base case ($n = 2$):** $P(2)$ says that all Braun trees with at most 2 nodes are unique. There is only one tree with one node, and it's also a Braun tree. There are two trees with two nodes and only one of them is Braun tree (you can draw them to see for yourself). So, Braun trees with at most 2 nodes are unique.
- **inductive step ($n \geq 2$):** Assume that $P(n)$ is true (that is our inductive hypothesis). Now, we have to prove that $P(n + 1)$ is also true, and we may use the inductive hypothesis in the proof. So, we have to prove that for all $1 \leq k \leq n + 1$, there is exactly one Braun tree with k nodes. For $k \leq n$, that follows directly from the inductive hypothesis.

Now, let $T = (V, E)$ be a Braun tree with $n + 1$ nodes. Let $T_L = (V_L, E_L)$ and $T_R = (V_R, E_R)$ be its left and right subtrees (the subtrees rooted in the left and right son of T 's root). There are now two cases to consider:

- $n + 1 = 2k$ ($n + 1$ is even). Then, it must be that $|V_L| = k$ and $|V_R| = k - 1$ by definition of the Braun tree. Since both k and $k - 1$ are less than n , we can invoke the inductive hypothesis, by which $P(k)$ and $P(k - 1)$ are both true. So, there is exactly one Braun tree with k nodes and exactly one Braun tree with $k - 1$ nodes. These trees are T_L and T_R . Since these are T 's respective left and right subtrees, T is also unique. So, there is only one Braun tree with $n + 1$ nodes, when $n + 1$ is even.

- $n + 1 = 2k + 1$ ($n + 1$ is odd). Then, $|V_L| = k$ and $|V_R| = k$, by definition of the Braun tree. The same argument as in the “even” case can now be applied, to yield that there is exactly one Braun tree with $n + 1$ nodes, when $n + 1$ is odd.

So, now we know that $P(n) \Rightarrow P(n + 1)$ for any $n \geq 2$. By principle of mathematical induction, $(\forall n \geq 2)P(n)$.

Two sidenotes:

- We had to “start” our induction from 2 instead of 1. That is because if we started from 1, then the inductive step would be more difficult to prove, since the proof given above would fail for $n = 1$ (proving $P(1) \Rightarrow P(2)$). Namely, the right subtree of the Braun tree with 2 nodes does not exist.
- This proof would have been considerably simpler if we used *total induction*. The difference between total induction and the one we did in labs is the inductive step:

$$((\forall k \leq n)P(k)) \Rightarrow P(n + 1),$$

for any $n \geq 1$. So, instead of using only one inductive hypothesis, $P(n)$, you can use all $P(1), P(2), \dots, P(n)$. That form of induction is not any more powerful than the “regular” induction, but is sometimes easier to use. Try to solve this exercise using the total induction.

Exercise 2: Another proof by induction. This time, we will use a variation of total induction. The variation is that we are allowed to have more base cases, if they are chosen carefully enough (explained later).

Let $P(n), n \geq 0$ be a predicate such that $P(n)$ is true if and only if $Fib(n) = (\phi^n - \psi^n)/\sqrt{5}$. Obviously, if we prove $P(n)$ for all $n \geq 0$, then we have proven the statement of the exercise.

- **base cases** ($n = 0, n = 1$): By simple calculation, it can be verified that $Fib(0) = (\phi^0 - \psi^0)/\sqrt{5}$ and $Fib(1) = (\phi^1 - \psi^1)/\sqrt{5}$. So, $P(0)$ and $P(1)$ are both true.
- **inductive step** ($n \geq 1$): We want to prove $P(n + 1)$, that is, we want to prove $Fib(n + 1) = (\phi^{n+1} - \psi^{n+1})/\sqrt{5}$. By definition of Fib , $Fib(n + 1) = Fib(n) + Fib(n - 1)$. Now, by inductive hypothesis, we know that

$$Fib(n) = (\phi^n - \psi^n)/\sqrt{5}$$

and

$$Fib(n - 1) = (\phi^{n-1} - \psi^{n-1})/\sqrt{5}.$$

Also, it’s easy to verify that $\phi^2 = \phi + 1$ and $\psi^2 = \psi + 1$ (by simple calculation). So,

$$Fib(n + 1) = (\phi^n - \psi^n)/\sqrt{5} + (\phi^{n-1} - \psi^{n-1})/\sqrt{5}$$

$$\begin{aligned}
&= ((\phi^n + \phi^{n-1}) + (\psi^n + \psi^{n-1}))/\sqrt{5} \\
&= (\phi^{n-1}(\phi + 1) + \psi^{n-1}(\psi + 1))/\sqrt{5} \\
&= (\phi^{n-1} \cdot \phi^2 + \psi^{n-1} \cdot \psi^2)/\sqrt{5} \\
&= (\phi^{n+1} + \psi^{n+1})/\sqrt{5}
\end{aligned}$$

So, we have proven that $Fib(n+1) = (\phi^{n+1} + \psi^{n+1})/\sqrt{5}$, that is, we have proven $P(n+1)$.

Now, by principle of mathematical induction, $P(n)$ is true for all $n \geq 0$.

Sidenote (“careful choice” of the base cases): If we, for example, chose only $n = 0$ to be our base case (that is, if we proved only $P(0)$ instead of both $P(0)$ and $P(1)$), then the inductive step would fail for $n = 1$. Can you explain why?

Exercise 3: Here is how `(+1 4 5)` is evaluated:

```

(+1 4 5)
(inc (+1 3 5))
(inc (inc (+1 2 5)))
(inc (inc (inc (+1 1 5))))
(inc (inc (inc (inc (+1 0 5)))))
(inc (inc (inc (inc 5))))
(inc (inc (inc 6)))
(inc (inc 7))
(inc 8)
9

```

Here is how `(+2 4 5)` is evaluated:

```

(+2 4 5)
(+2 (dec 4) (inc 5))
(+2 3 6)
(+2 (dec 3) (inc 6))
(+2 2 7)
(+2 (dec 2) (inc 7))
(+2 1 8)
(+2 (dec 1) (inc 8))
(+2 0 9)
9

```

So, `+1` is not tail-recursive, while `+2` is.