Computing Science 204 (B1)

Analysis of the Recursive Fibonacci Program Fib1

A recursive algorithm for computing the Fibonacci numbers $0,\ 1,\ 1,\ 2,\ 3,\ 5,\ 8, \cdots$ is given below.

```
int Fib1(int n)
{
   if (n < 2)
       return(n);
   else
       return(Fib1(n-1)+Fib1(n-2));
}
```

For $n \geq 0$, let $T(n)$ be the number of calls made to `Fib1` when calculating the n^{th} Fibonacci number F_n, that is, $T(n)$ is the number of nodes in the recursive calling tree. In this note we will

(a) find a recurrence relation together with initial conditions satisfied by $T(n)$ for $n \geq 2$.

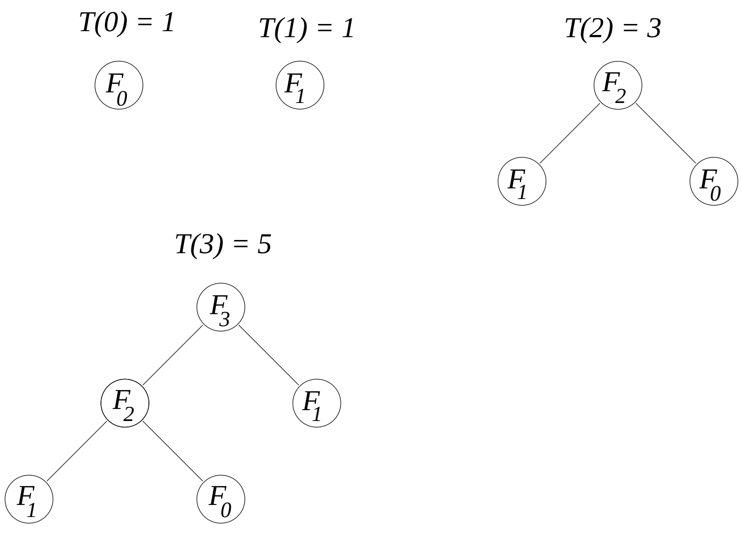
(b) solve this recurrence relation to get $T(n),\ n \geq 0$.

(c) show that $T(n) > \left(\dfrac{1+\sqrt{5}}{2}\right)^{n-1}$ for $n \geq 2$.

(d) show that $T(n) > 2^{\frac{n}{2}}$ for $n \geq 2$.

Solution:

(a) The recursive calling trees for the algorithm `Fib1` for the first few values of n are shown below.



If $T(n)$ is the number of calls to `Fib1` that are made when calculating F_n, then $T(n)$ satisfies the following recurrence relation and initial conditions:

$$T(n) = T(n-1) + T(n-2) + 1, \quad n \geq 2$$

$$T(0) = 1$$

$$T(1) = 1.$$

(b) If we define $b_n = \dfrac{T(n)+1}{2}$, for $n \geq 0$, then it is easy to see that b_n satisfies the recurrence relation

$$b_n = b_{n-1} + b_{n-2}, \quad n \geq 2$$
$$b_0 = 1$$
$$b_1 = 1,$$

but this is the recurrence relation satisfied by F_{n+1}, the $(n+1)^{\text{st}}$ Fibonacci number. Since there is only one sequence satisfying this recurrence relation with the given initial values, then we must have $b_n = F_{n+1}$ for all $n \geq 0$, therefore, $T(n) = 2F_{n+1} - 1$ for $n \geq 0$.

(c) Let

$$\alpha = \frac{1+\sqrt{5}}{2},$$

since $T(n) = 2F_{n+1} - 1 = F_{n+1} + F_{n+1} - 1$, and $F_{n+1} > 1$ for $n \geq 2$, then in order to show that $T(n) > \alpha^{n-1}$ for all $n \geq 2$, all we need to show is that $F_{n+1} \geq \alpha^{n-1}$ for all $n \geq 2$. The proof is by induction on n. We will show in fact that $F_{n+1} \geq \alpha^{n-1}$ for all $n \geq 1$.

For $n = 1$, $F_2 = F_{1+1} = 1 = \alpha^0 = \alpha^{1-1}$, and the result is true for $n = 1$. Now suppose the result is true for all positive integers k, with $1 \leq k \leq n$, then

$$F_{n+2} = F_{n+1} + F_n \geq \alpha^{n-1} + \alpha^{n-2} = \alpha^{n-2}(1+\alpha) = \alpha^{n-2}\left(1 + \frac{1+\sqrt{5}}{2}\right)$$
$$= \alpha^{n-2}\left(\frac{3+\sqrt{5}}{2}\right) = \alpha^{n-2}\left(\frac{1+\sqrt{5}}{2}\right)^2 = \alpha^{n-2}\alpha^2 = \alpha^n,$$

and the result is true for $n+1$ also. Therefore, by the principle of mathematical induction, $F_{n+1} \geq \alpha^{n-1}$ for all $n \geq 1$, and this implies that $T(n) > \alpha^{n-1}$ for all $n \geq 2$.

(d) Again, since $T(n) = 2F_{n+1} - 1 = F_{n+1} + F_{n+1} - 1$ and $F_{n+1} > 1$ for all $n \geq 2$, in order to show that $T(n) > 2^{\frac{n}{2}}$, for all $n \geq 2$, we only need to show that $F_{n+1} \geq 2^{\frac{n}{2}}$ for $n \geq 2$. Again, the proof is by induction on n.

For $n = 2$, $F_{2+1} = F_3 = 2 = 2^1 = 2^{\frac{2}{2}}$, and the result is true for $n = 2$. Now suppose the result is true for all positive integers k, with $2 \leq k \leq n$, then

$$F_{n+2} = F_{n+1} + F_n \geq 2^{\frac{n}{2}} + 2^{\frac{n-1}{2}}$$
$$= 2^{\frac{n-1}{2}}(1+\sqrt{2})$$
$$\geq 2 \cdot 2^{\frac{n-1}{2}} = 2^{\frac{n+1}{2}},$$

and the result is true for $n+1$ also. Therefore, by the principle of mathematical induction, $F_{n+1} \geq 2^{\frac{n}{2}}$ for all $n \geq 2$, and this implies that $T(n) > 2^{\frac{n}{2}}$ for all $n \geq 2$.