Analysis of Algorithms

Problem 6-1

Use the operator technique to obtain a solution to the following recurrence:

$$a_{n+2} = (n+2)a_{n+1} - na_n + n,$$

with $a_0 = 0 = a_1$.

Solution

We may rewrite the recurrence in operator notation as follows:

$$(\mathbf{E}^2 - (n+2)\mathbf{E} + n)a_n = n.$$

The polynomial $\mathbf{E}^2 - (n+2)\mathbf{E} + n$ can be factored into $(\mathbf{E} - 1)(\mathbf{E} - n)$, using the identity $\mathbf{E}n = (n+1)\mathbf{E}$. We now let $b_n = (\mathbf{E} - n)a_n$, for $n \geq 0$, and solve for b_n. We have $(\mathbf{E} - 1)b_n = n$, which written in full, amounts to $b_{n+1} = b_n + n$, where $n \geq 0$ and $b_0 = a_1 = 0$. This gives

$$b_n = b_0 + \frac{n(n-1)}{2} = \frac{n(n-1)}{2}.$$

We now solve for a_n. We have $(\mathbf{E} - n)a_n = b_n$ and this may be written out as $a_{n+1} = na_n + \frac{n(n-1)}{2}$. Divide by $n!$ on both sides to obtain:

$$\frac{a_{n+1}}{n!} = \frac{a_n}{(n-1)!} + \frac{1}{2(n-2)!}.$$

Let $c_{n+1} = a_{n+1}/n!$ for $n \geq 0$. Then we may rewrite the above recurrence as

$$c_{n+1} = c_n + \frac{1}{2(n-2)!},$$

where $n \geq 2$. Note that $c_2 = a_2/1! = a_2 = 0$. This gives us $c_{n+1} = \sum_{p=0}^{n-2} \frac{1}{2p!}$ for $n \geq 2$. Since $a_{n+1} = n!c_{n+1}$, we have

$$a_{n+1} = n! \sum_{p=0}^{n-2} \frac{1}{2p!},$$

for $n \geq 2$. The initial conditions are $a_0 = a_1 = a_2 = 0$.

Problem 6-2

Suppose that the following program is called from a `main` function and let a_n be the number of calls to $f(n,k)$.

1. Determine the recurrence of a_n and express it in terms of the shift operator $\mathbf{E}$.
2. Reduce the degree of the recurrence by factorizing the operator expression. Solve the newly obtained recurrence b_n.

```
int f(n, k) {
   int sum = k;
   if(n > 1) {
      for(int i = 0; i <= n; ++i){
         sum += f(n - 1, i);
         for(int j = 0; j < n; ++j){
          sum += f(n - 2, i + j);
            sum += f(n - 2, i + 2 * j);
         }
      }
   } else {
       sum = n;
   }
   return sum;
}
```

Solution

First note that the second parameter in the function does not influence the number of recursive calls. For $n \in \{0, 1\}$, the function f is called once and hence $a_0 = a_1 = 1$. For $n \geq 2$, there is one call to $f(n)$ and inside the function there are $n + 1$ calls to $f(n - 1)$ and $2n(n + 1)$ calls to $f(n - 2)$. Therefore a_n may be written as

$$a_n = (n + 1)a_{n-1} + 2n(n + 1)a_{n-2} + 1,$$

which in operator notation can be written as

$$(\mathbf{E}^2 - (n + 1)\mathbf{E} - 2n(n + 1))a_{n-2} = 1.$$

The operator $\mathbf{E}^2 - (n+1)\mathbf{E} - 2n(n+1)$ can be factorized as $(\mathbf{E} - 2(n+1))(\mathbf{E} + n)$. We let $(\mathbf{E} + n)a_{n-2} = b_{n-2}$, where $n \geq 2$. Note that $b_0 = a_1 + 2 \cdot a_0 = 3$. Our first reduced recurrence is therefore: $(\mathbf{E} - 2(n + 1))b_{n-2} = 1$, which may be rewritten as:

$$b_n - 2(n + 2)b_{n-1} = 1 \text{ where } n \geq 1 \text{ and } b_0 = 3.$$

Dividing both sides by $2^n \cdot (n + 2)!$, we obtain:

$$\frac{b_n}{2^n \cdot (n + 2)!} = \frac{b_{n-1}}{2^{n-1} \cdot (n + 1)!} + \frac{1}{2^n \cdot (n + 2)!}.$$

Setting $c_n = \frac{b_n}{2^n \cdot (n+2)!}$, where $n \geq 0$, we obtain that:

$$c_n = \frac{b_0}{2!} + \sum_{j=1}^{n} \frac{1}{2^j \cdot (j + 2)!} = \frac{3}{2} + \sum_{j=1}^{n} \frac{1}{2^j \cdot (j + 2)!}.$$

Therefore b_n is given by:

$$b_n = 2^n \cdot (n + 2)! \cdot \left(\frac{3}{2} + \sum_{j=1}^{n} \frac{1}{2^j \cdot (j + 2)!} \right).$$

Homework Assignment 6-1 (10 Points)

Provide a closed form for a_n from Problem 6-2 (up to initial conditions), using the solution obtained for b_n in class.

Homework Assignment 6-2 (10 Points)

Solve the following recurrence by reducing its degree.

$$\begin{aligned} a_0 &= 8000 \\ a_1 &= \frac{1}{2} \\ a_{n+2} + a_{n+1} - n^2 a_n &= n! \end{aligned}$$