The Hiring Problem Revisited

Recall early in this course we defined and analyzed the hiring problem. To summarize, in the hiring problem, you are seeking to hire an office assistant, and you are using an employment agency to help. Because you are using the employment agency, certain charges will need to be paid. If you accept a candidate to be interviewed, you are required to pay the agency c_i. If you actually hire someone that was interviewed, you need to pay the agency c_h. The algorithm we considered is as follows:

Algorithm 1 Hire Assistant

```
HireAssistant(n)
  best ← 0;
  for i ← 1 to n do
    interview candidate i;
    if candidate i is better than candidate best then
      best ← i;
      hire candidate i;
    end if;
  end for;
```

Recall that, on average, we only hire $O(\ln n)$ people with a total expected cost of $O(c_h \ln n)$. Even with this expected-cost calculation, the algorithm we considered was still deterministic. Specifically, for any given sequence of candidates provided as input, those hired would always be the same.

Now let's consider a "randomized" version of the hiring algorithm. Specifically, suppose the first thing we do in the algorithm is take the input sequence and randomize the associated permutation (and thereby the order) of the candidates.

Algorithm 2 Randomized Hire Assistant

```
RandomHireAssistant(n)
  randomly permute the list of candidates
  best ← 0;
  for i ← 1 to n do
    interview candidate i;
    if candidate i is better than candidate best then
      best ← i;
      hire candidate i;
    end if;
  end for;
```

In this case, we have introduced a randomization element in the algorithm itself rather than relying on an input distribution. Thus, for the same sequence of candidates provided to the algorithm, a different set of people hired would be experienced on each run of the algorithm. The advantage to this approach is that no "adversary" can force the algorithm to perform badly. Only an unlucky shuffle of the input sequence would cause this.

Lemma: The expected hiring cost of the algorithm RandomizedHireAssistant is $O(c_k \ln n)$.

Proof: After permuting the array, we are now faced with a situation identical to that used with the probabilistic analysis of HireAssistant. The only difference is that the randomization occurs in the algorithm itself. QED