## Section 5.1 – The hiring problem

5.1-1 Show that the assumption that we are always able to determine which candidate is best, in line 4 of procedure Hire-Assistant, implies that we know a total order on the ranks of the candidates.

> Let $A$ be the set of candidates in random order and $R$ the binary relation "is better than or equal" on the set $A$. $R$ is a total order if
>
> (a) $R$ is **reflexive**. That is, $a \; R \; a \;\; \forall \, a \in A$;
>
> (b) $R$ is **antisymmetric**. That is, $a \; R \; b$ and $b \; R \; a$ imply $a = b$;
>
> (c) $R$ is **transitive**. That is, $a \; R \; b$ and $b \; R \; c$ imply $a \; R \; c$;
>
> (d) $R$ is a **total relation**. That is, $a \; R \; b$ or $b \; R \; a \;\; \forall \, a, b \in A$.
>
> The above properties are necessary because
>
> (a) if two different candidates have the same qualification, it is necessary so that they can be compared;
>
> (b) if both $a$ is "better than or equal" than $b$ and $b$ is "better than or equal" than $a$ and they qualifications are not equal, we would not be able to choose one of them and still be hiring "the best candidate we have seen so far";
>
> (c) if we hire $b$ because he is "better than or equal" than $a$ and then we hire $c$ because he is "better than or equal" than $b$ and $c$ is not "better than or equal" than $a$, we are not hiring "the best candidate we have seen so far";
>
> (d) if the $R$ is not a total relation, we would not be able to compare any two candidates.

5.1-2 ($\star$) Describe an implementation of the procedure Random(a, b) that only makes calls to Random(0, 1). What is the expected running time of your procedure, as a function of a and b?

> The pseudocode is stated below.
>
> ```
> RandomInterval(a, b)
>   flips = ⌈lg(b − a)⌉
>   count = ∞
>   while count > b do
>     count = 0
>     for i = 1 to flips do
>       count = count + (2^(i−1) · Random(0, 1))
>   return count + a
> ```
>
> The expected running time is
>
> $$\underbrace{2^{\lceil \lg(b-a) \rceil}/(b-a)}_{\text{while loop}} \cdot \underbrace{\lceil \lg(b-a) \rceil}_{\text{for loop}} < 2 \cdot \lceil \lg(b-a) \rceil,$$
>
> where the last inequality is valid since $1 \leq 2^{\lceil \lg(b-a) \rceil}/(b-a) < 2$.

5.1-3 ($\star$) Suppose that you want to output 0 with probability 1/2 and 1 with probability 1/2. At your disposal is a procedure Biased-Random, that outputs either 0 or 1. It outputs 1 with some probability $p$ and 0 with probability $1 - p$, where $0 < p < 1$, but you do not know what $p$ is. Give an algorithm that uses Biased-Random as a subroutine, and returns an unbiased answer, returning 0 with probability 1/2 and 1 with probability 1/2. What is the expected running time of your algorithm as a function of $p$?

> The pseudocode is stated below.
>
> ```
> Random()
>   while 1 do
>     r1 = Random(0, 1)
>     r2 = Random(0, 1)
>     if r1 ≠ r2 then
>       return r1
> ```
>
> The expected running time is
>
> $$\frac{1}{\underbrace{(1-p)p}_{(r_1, r_2) = (0,1)} + \underbrace{p(1-p)}_{(r_1, r_2) = (1,0)}} \cdot 1 = \frac{1}{2p(1-p)}.$$

## Section 5.2 – Indicator random variables

5.2-1 In HIRE-ASSISTANT, assuming that the candidates are presented in a random order, what is the probability that you hire exactly one time? What is the probability that you hire exactly $n$ times?

Since the initial dummy candidate is the least qualified, HIRE-ASSISTANT will always hire the first candidate. It hires exacly one time when the best candidate is the first to be interviewed. Thus, the probability is $1/n$. To hire exactly $n$ times, the candidates has to be in increasing order of quality. Since there are $n!$ possible orderings (each one with equal probability of happening), the probability is $1/n!$.

5.2-2 In HIRE-ASSISTANT, assuming that the candidates are presented in a random order, what is the probability that you hire exactly twice?

The first candidate is always hired, thus the best qualified candidate cannot be the first to be interviewed. Also, among all the candidates that are better qualified than the first candidate, the best candidate must be interviewed first. Otherwise, a third candidate will be hired between them. Now assume that the first candidate to be interviewed is the $i$-th best qualified, for $i = 2, \ldots, n$. This occurs with a probability of $1/n$. To hire exactly twice, the best candidate must be the first to be interviewed among the $i-1$ candidates that are better qualified than candidate $i$. This occurs with a probability of $1/(i-1)$. Thus, the probability of hiring exactly twice is

$$\sum_{i=2}^{n} \frac{1}{n}\frac{1}{i-1} = \frac{1}{n}\sum_{i=1}^{n-1}\frac{1}{i} = \frac{1}{n}(\lg(n-1) + O(1)).$$

5.2-3 Use indicator random variables to compute the expected value of the sum of $n$ dice.

Let $X_i$ be an indicator random variable of a dice coming up the number $i$. We have $\Pr\{X_i\} = 1/6$. Let $X$ be a random variable denoting the result of throwing a dice. Then

$$\mathrm{E}[X] = \sum_{i=1}^{6} i \cdot \Pr\{X_i\} = \sum_{i=1}^{6} i \cdot \frac{1}{6} = \frac{1}{6}\sum_{i=1}^{6} i = \frac{1}{6}\frac{6\cdot 7}{2} = 3.5.$$

By linearity of expectations, the expected value of the sum of $n$ dice is the sum of the expected value of each dice. Thus,

$$\sum_{i=1}^{n} \mathrm{E}[X] = \sum_{i=1}^{n} 3.5 = 3.5 \cdot n.$$

5.2-4 Use indicator random variables to solve the following problem, which is known as the ***hat-check problem***. Each of $n$ customers gives a hat to a hat-check person at a restaurant. The hat-check person gives the hats back to the customers in a random order. What is the expected number of customers who get back their own hat?

Let $X_i$ be an indicator random variable of customer $i$ getting back his own hat. We have

$$\Pr\{X_i\} = \mathrm{E}[X_i] = 1/n.$$

Let $X$ be a random variable denoting the number of customers who get back their own hat. Then

$$X = X_1 + X_2 + \cdots + X_n,$$

which implies

$$\begin{aligned}\mathrm{E}[X] &= \mathrm{E}\left[\sum_{i=1}^{n} X_i\right] \\ &= \sum_{i=1}^{n} \mathrm{E}[X_i] \\ &= \sum_{i=1}^{n} \frac{1}{n} \\ &= 1.\end{aligned}$$

5.2-5 Let $A[1, \ldots, n]$ be an array of $n$ distinct numbers. If $i < j$ and $A[i] > A[j]$, then the pair $(i, j)$ is called an ***inversion*** of $A$. (See Problem 2-4 for more on inversions.) Suppose that the elements of $A$ form a uniform random permutation of $\langle 1, 2, \ldots, n \rangle$. Use indicator random variables to compute the expected number of inversions.

Let $X_{ij}$ be an indicator random variable for the event that the pair $(i, j)$ is inverted. Since $A$ forms a uniform random permutation, we have

$$\Pr\{X_{ij}\} = \Pr\{\overline{X_{ij}}\} = 1/2,$$

which implies

$$\mathrm{E}[X_{ij}] = 1/2.$$

Let $X$ be a random variable denoting the number of inversions of $A$. Since there are $\binom{n}{2}$ possible pairs on $A$, each with probability $1/2$ of being inverted, we have

$$\mathrm{E}[X] = \binom{n}{2}\frac{1}{2} = \frac{n!}{2! \cdot (n-2)!}\frac{1}{2} = \frac{n(n-1)}{4}.$$