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
>
> $$\frac{1}{\underbrace{(1-p)p}_{(r_1, r_2) = (0,1)} + \underbrace{p(1-p)}_{(r_1, r_2) = (1,0)}} \cdot 1 = \frac{1}{2p(1-p)}.$$