CPSC 351 — Tutorial Exercise #16
Additional Practice Problems

This problem will not be discussed during the tutorial, and solutions for this problem will not be made available. It can be used as a "practice" problem that can help you practice skills considered in the lecture presentation for Lectures #18, or in Tutorial Exercise #16.

Problems To Be Solved

Once again, consider the "binary search tree" experiment. For $n \geq 1$, let Ω_n be the sample space that has been described for this experiment, so that

$$\Omega_1 = \{(1)\},$$

$$\Omega_2 = \{(1,2),(2,1)\},$$

and

$$\Omega_3 = \{(1,2,3),(1,3,2),(2,1,3),(2,3,1),(3,1,2),(3,2,1)\}.$$

It will be useful, for later, to ***define*** a sample space Ω_0 (for the case $n = 0$) to be

$$\Omega_0 = \{()\}$$

— as set of size *one* whose outcome is represented as by the empty sequence, "$()$". Then $|\Omega_0| = 1$ and, for every positive integer n, $|\Omega| = n!$.

For each integer n such that $n \geq 0$, let T_n be the set of ***binary search trees*** storing the set

$$\{1, 2, \ldots, n\}$$

— so that T_0 is the set of size one, including the empty tree, while T_1 is a set of size one as well, since T_1 includes a single binary search tree, with size one, storing "1" at the root.

Thus $|\Omega_0| = |T_0| = 1$, $|\Omega_1| = |T_1|$ and, indeed, $|\Omega_2| = |T_2| = 2$.

However, if you completed the second problem in this exercise then you may have noticed that there are only ***five*** binary search trees, with size three, that stores the numbers 1, 2, and 3, so that

$$|T_3| = 5 < 6 = |\Omega_3|.$$

Two of the outcomes in the same space Ω_3 correspond to the same binary search tree in T_3.

In general — since the binary search tree being constructed is determined by the order in which numbers are inserted into an initially empty tree (that is, determined by the outcome in Ω_n), each binary search tree in T_n corresponds to an ***event*** for the experiment with sample space Ω_n, but not necessarily an ***elementary event***.

With that noted, we can now consider a ***second binary search tree experiment*** that depends on a non-negative integer n, in which one "picks a binary search tree that stores the set $\{1, 2, \ldots, n\}$ — so that the ***sample space*** is T_n instead of Ω_n. While all the "events" that make sense for the second experience also make sense for the first, the opposite is not true. For example, the event "1 is the second number inserted into the tree" is a well-defined event when $n = 3$, for the first experiment — and corresponds to the subset

$$\{(2, 1, 3), (3, 1, 2)\}$$

of Ω_3 — but it is not a well-defined event for the second experiment, with sample space T_3, at all!

With that noted, we will now consider a ***third binary search tree experiment*** — which is, in a sense, somewhere "between"' these first two. Once again, let n be a positive integer. Suppose that we construct a binary search tree, storing the set $\{1, 2, \ldots, n\}$, by doing the following:

- First, pick the number (between 1 and n that must be stored at the root.
- Suppose that the number chosen was i. Then the left subtree must be a binary search tree storing the numbers from 1 to $i-1$. Use the ***first experiment*** — with $i-1$ replacing n — to choose the left subtree of the root.
- The right subtree must be a binary search tree with size $n-i$ — which stores the numbers

  $$i + 1, i + 2, \ldots, n$$

  instead of the numbers

  $$1, 2, \ldots, n - i.$$

  Suppose we use the ***first experiment*** to construct the right subtree as well, adding i to the number stored at each node, so that the right subtree stores the numbers it should.

For $n \geq 1$, the ***sample space*** Λ_n, corresponding to this, could consist of a set of ordered triples

$$(i, \alpha_L, \alpha_R)$$

where i is an integer such that $1 \leq i \leq n$ (the value stored at the root), $\alpha_L \in \Omega_{i-1}$, and $\alpha_R \in \Omega_{n-i}$.

Thus

$$\Lambda_1 = \{(1, (), ())\}$$

(a set with size one),

$$\Lambda_2 = \{(1, (), (1)), (2, (1), ())\}$$

(a set with size two), and

$$\Lambda_3 = \{(1, (), (1, 2)), (1, (), (2, 1)), (2, (1), (1)), (3, (1, 2), ()), (3, (2, 1), ())\}$$

(a set with size five). As noted above (in words),

$$\Lambda_n = \{(i, \alpha_L, \alpha_R) \mid i \in \mathbb{Z}, 1 \leq i \leq n, \alpha_L \in \Omega_{i-1}, \text{ and } \alpha_R = \Omega_{n-i}\}.$$

It follows that, for $n \geq 1$,

$$|\Lambda_n| = \sum_{i=1}^{n} |\Omega_{i-1}| \times |\Omega_{n-i}| = \sum_{i=1}^{n} (i-1)! \times (n-i)!.$$

In general, $|T_n| \leq |\Lambda_n| \leq |\Omega_n|$ — and $|T_n| < |\Lambda_N| < |\Omega_n|$ if $n \geq 4$.

1. For $n \geq 1$, consider the function $\mathsf{P} : \Lambda_n \to \mathbb{R}$ such that

   $$\mathsf{P}((i, \alpha_L, \alpha_R)) = \frac{1}{n} \times \frac{1}{(i-1)!} \times \frac{1}{(n-i)!}$$

   for every integer i such that $1 \leq i \leq n$, and for all $\alpha_L \in \Omega_{i-1}$ and for all Ω_{n-i} — where we consider $0!$ to be equal to 1.

   Prove that this defines a ***probability distribution*** for the sample space Λ_n (and for the third binary search tree experiment).

We will see, later, that the ***third experiment***, with this probability distribution, is useful when studying properties of binary search trees that are generated using the ***first experiment*** — so this will be continued in one or more later exercises.