Combinatorics (DS I) - Sheet 6

Exercise 6.1.

Let B_n be the complete binary tree of depth n. It is defined recursively, we set B_0 to be a single vertex, which we call its root, then B_n is obtained by placing two copies of B_{n-1} next to each other, drawing a new root above them and connecting it to the roots of the two copies of B_{n-1}. Let $S_{n,k}$ be the star with n legs of length k. It is defined by placing n paths of k vertices each, then placing a root and connecting the root to an extremal vertex of each path. See Figure 1 for an illustration.

(a) How many subtrees of B_n contain its root?

(b) How many subtrees of $S_{n,k}$ contain its root?

(c) Suggest an algorithm to compute the number of subtrees of a rooted tree which contain the root. What is the runtime of your algorithm?

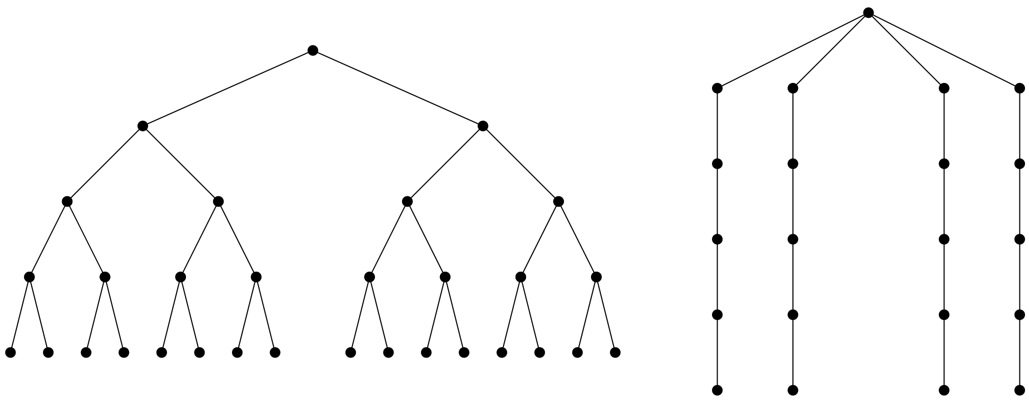

Figure 1: On the left, the complete binary tree of depth 4. On the right, the star with 4 legs of length 5.

Exercise 6.2.

Let $F(z) = \sum \frac{B(n)}{n!} z^n$ be the exponential generating function of the Bell numbers $B(n)$.

(a) Find the function f such that

$$F'(z) = f(z)F(z).$$

(b) Solve the differential equation and deduce a closed form for $F(z)$.

Exercise 6.3.

For $k \in [n]$, how many sequences $(T_1, \ldots, T_k)$ are there with $T_0 = \emptyset \subset T_1 \subset \ldots \subset T_{k-1} \subset [n] = T_k$?

(a) Find the nice short formula.

(b) By differentiating by the number of equalities, find a formula involving the Stirling numbers of the second kind.

Exercise 6.4.

The binary reflected Gray code of order n (BRGC(n)) is a cyclic listing of all bitstrings of length n, i.e. elements of $\{0,1\}^n$, such that neighbouring strings differ in only one bit. It is defined recursively as follows: BRGC(1)=$0, 1$, and BRGC(n) is obtained from BRGC($n-1$) as follows: Let P be BRGC($n-1$) with a 0 appended to every bitstring and let Q be the reverse of BRGC($n-1$) with a 1 appended to every bitstring. Then BRGC(n) is the concatenation of P and Q.

(a) Prove that BRGC(n) really lists all bitstrings exactly once and that neighbouring strings really differ in only one bit.

(b) Given some $b \in \{0,1\}^n$, how can you find the position of b in BRGC(n) efficiently?

(c) Given some $k \in [2^n]$, how can you find the bitstring at position k of BRGC(n) efficiently?

(d) Given some $b \in \{0,1\}^n$, how can you find the bitstring following it in BRGC(n) efficiently?

0000	0011
1000	1011
1100	1111
0100	0111
0110	0101
1110	1101
1010	1001
0010	0001

Figure 2: The binary reflected Gray code for bitstrings of length 4, read left column top to bottom then right column top to bottom. Note how the columns contain the code for strings of length 3.

Exercise 6.5.

Given a graph G, a spanning tree is a subtree of the graph which connects all the vertices. The ladder of length n is obtained by connecting two paths of length n by a matching as in Figure 3. The pencil of length n is obtained from the ladder of length $n-1$ by adding an extra vertex and connecting it to the ends of the paths, as in Figure 3. We want to compute the number of spanning trees of the ladder of length n.

(a) Let $L(n)$ and $P(n)$ be the number of spanning trees in the ladder and pencil of length n respectively. Find recursive relations for $L(n)$ and $P(n)$ in terms of each other.

(b) A spanning tree either contains the rightmost edge or it does not.

(c) Use the recurrences to obtain the generating function of $L(n)$ and $P(n)$ in terms of each other. Then substitute to obtain the generating function of $L(n)$ as a rational function.

(d) Use the generating function to obtain a closed form for $L(n)$.

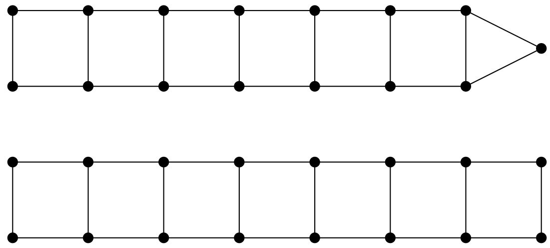

Figure 3: The pencil and ladder of length 8.

Bonus Exercise

A merchant has 12 gold coins, but one of them is a fake. The weight of the fake is different from the weight of the real coins, and all real coins weigh the same. You are given an old scale with which you can compare the weight of two sets of coins. Using the scale only 3 times, can you figure out which is the fake?

List of hints

- Two objects on this sheet are counted by the same numbers.
- Think about exactly what information you get when you weigh two sets of coins.
- Even the general case has a recursive definition.
- Go through the string right to left and keep track if you are in a reflected part or not.
- Use the recursion
$$B(n+1) = \sum_{k=0}^{n} \binom{n}{k} B(n-k).$$
- Prove
$$\sum_n \left(\sum_{k=0}^{n} \frac{B(k)}{k!} \frac{1}{(n-k)!} \right) x^n = \left(\sum_n \frac{B(n)}{n!} x^n \right) \left(\sum_n \frac{x^n}{n!} \right).$$
- A spanning tree either contains the rightmost edge, or not.
- Look at the left and right side individualy.