Counting Well-Formed Parenthesizations Easily

Pekka Kilpeläinen
University of Eastern Finland
School of Computing, Kuopio

August 20, 2014

Abstract

It is well known that there is a one-to-one correspondence between ordered trees of \(n+1\) nodes, forests of \(n\) nodes, binary trees of \(n\) nodes, and well-formed parenthesizations with \(n\) opening and \(n\) closing parentheses, which leads to the classic result that the number of objects in each of these collections is given by the \(n\)th Catalan number \(C_n\). The closed form of \(C_n\) is typically derived as an application of generating functions. This note presents a simple combinatorial argument for the fact that the number of well-formed parenthesizations of \(n\) opening and \(n\) closing parentheses is \(\frac{1}{n+1} \binom{2n}{n}\).

Key words: Well-formed parenthesization, Dyck word, Catalan number, counting, Raney’s lemma

1 Introduction

It is well known that the \(n\)th Catalan number \(C_n\) gives the number of different ordered trees of \(n+1\) nodes, forests of \(n\) nodes, binary trees of \(n\) nodes, and well-formed parenthesizations with \(n\) opening and \(n\) closing parentheses. Catalan numbers have surprisingly many additional combinatorial interpretations. Stanley mentions about 70 of them in his book (1999). Together with an addendum published on his web site\(^2\) the number of combinatorial interpretations of \(C_n\) is 207. Koshy (2009, Chap. 5) provides a historical introduction to Catalan numbers.

The closed form \(\frac{1}{n+1} \binom{2n}{n}\) of the \(n\)th Catalan number \(C_n\) is typically derived as follows (See, e.g., Knuth [1973, pp. 388-389] or Graham et al.\(^1\)).

\(^1\) The overall idea of this proof was given by late Derick Wood in personal communication at the University of Waterloo, Ontario, in 1994.

\(^2\) www-math.mit.edu/~rstan/ec/
1990 pp. 343-344)): First a recurrence is derived for $C_n$, with

$$C_0 = 1$$

$$C_n = \sum_{i=0}^{n-1} C_i C_{n-1-i} \text{ for } n \geq 1 .$$

Then a quadratic equation is derived for a generating function $C(z)$ whose coefficients are $C_n$, and solved as

$$C(z) = \frac{1 - \sqrt{1 - 4z}}{2z} .$$

Using the binomial theorem and properties of binomial coefficients the closed form of $C(z)$ can be transformed back to a power series, whose coefficients finally give the result $C_n = \frac{1}{n+1} \binom{2n}{n}$.

Generating functions are a powerful tool to solve combinatorial enumeration problems; see (Graham et al., 1990, Chap. 7). Their drawback is that by manipulating formal power series one easily loses combinatorial intuition of the objects that are being studied. For this reason we present a simple combinatorial proof for the value of $C_n$. The argument is based on placing well-formed parenthesizations in correspondence with strings of parentheses which are straight-forward to count.

A string of $n$ opening parentheses ‘(’ and $n$ closing parentheses ‘)’ is a well-formed parenthesization (wfp) if each of its opening parentheses is matched by a unique closing parenthesis that follows it, and vice versa. The following are the five well-formed parenthesizations with three opening and three closing parentheses:

$$(((())), (()()), ()(())), (), ()()(), and ()()()$$

Well-formed parenthesizations consist of equally many opening and closing parentheses, but not all such strings are well-formed. In general, a string of $n$ opening and $n$ closing parentheses can be formed by choosing which $n$ out of the $2n$ available positions are occupied by an opening parenthesis, and placing a closing parenthesis at each of the remaining $n$ positions (or vice versa). Thus the total number of such strings is $\binom{2n}{n}$. For example, there are $\binom{6}{3} = 20$ different strings that consist of three opening and three closing parentheses. In addition to the above five of them which are well-formed parenthesizations, below are the remaining 15 which are not well-formed:

$$((()))(, (()())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, ))())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((())(, ))(()(), (())())(, )((.)
Table 1: Correspondence btw Dyck-1 words of length 7 and strings of 4 opening and 3 closing parentheses

<table>
<thead>
<tr>
<th>Dyck-1</th>
<th>Rotations, or strings with four ’(’ and three ’)’</th>
</tr>
</thead>
<tbody>
<tr>
<td>(((()))</td>
<td>(((()))), ((()))(, (())(((), ()))(((), ))((((), ), )(((), )))</td>
</tr>
<tr>
<td>((()))</td>
<td>((()))), (())(, (())(((), )))((((), ), )(((), )))</td>
</tr>
<tr>
<td>()(())</td>
<td>()(()), (())(, (())(((), )(((), ), ))((((), ), )(((), )))</td>
</tr>
<tr>
<td>()()()</td>
<td>()()()), (())(, (())(((), )(((), ), ))((((), ), )(((), )))</td>
</tr>
<tr>
<td>((())()</td>
<td>((())()), (())((, )()(((), )(((), ), ))((((), ), )(((), )))</td>
</tr>
</tbody>
</table>

Our proof for the number of well-formed parenthesizations is based on a correspondence between wfp’s and their rotations. For a string $w$ which starts by a prefix $\alpha$ that is followed by the suffix $\beta$, the strings $w = \alpha\beta$ and $\beta\alpha$ are rotations (aka cyclic shifts) of each other. When both $\alpha$ and $\beta$ are non-empty strings, the strings $\alpha\beta$ and $\beta\alpha$ are proper rotations of each other.

One complication for counting wfp’s in terms of their rotations is that some wfp’s have rotations which are identical. For example, ()()() has only two different rotations, which are ()()() and )()(). Another complication is that the rotations of some wfp’s coincide. For example, the wfp’s ()() and (()) have the following common set of rotations:

$$\{(()), (())(, ()()), (())(, ))(()(, )())\}$$

Surprisingly these complications disappear when we prefix the wfp’s by an extra opening parenthesis. We call the resulting strings Dyck-1 words. Let $C_n$ be the number of well-formed parenthesizations of length $2n$. Clearly the number of Dyck-1 words of length $2n + 1$ is also $C_n$. Let $A_n$ be the set of strings of $n + 1$ opening and $n$ closing parentheses. The size of $A_n$ is easily seen to be $\binom{2n+1}{n}$. As we’ll prove formally, the rotations of Dyck-1 words partition the set $A_n$ in $C_n$ equivalence classes each with $2n + 1$ members. This yields for the $n$th Catalan number its closed form

$$C_n = \frac{\binom{2n+1}{n}}{2n + 1} = \frac{\binom{2n}{n}}{n + 1}.$$

For example, Table 1 gives the correspondence between the five Dyck-1 words of length 7 and the $\binom{7}{3} = 35$ strings with four opening and three closing parentheses, which yields the number $35/7 = 5$ of wfp’s of length 6.

## 2 Proof

Next we justify the details of the above claim.
All strings that we consider in this note are over the alphabet \{(),\}. For a string \(w\) we use \(d(w)\) to denote the depth of string \(w\), which we define as the difference between the number of opening parentheses and the number of closing parentheses that occur in \(w\). For example, \(d(\varepsilon) = d(“()”) = d(“)(()”) = 0\), \(d(“((())”) = -2\), and \(d(“(()”) = 1\). Notice that the depth of a string is the sum of the depths of its subwords, that is, if \(w = uv\) then \(d(w) = d(u) + d(v)\). Well-formed parenthesizations can be characterized as those strings \(w\) which satisfy the conditions that \(d(w) = 0\) and \(d(\alpha) \geq 0\) for each prefix \(\alpha\) of \(w\). They are also known as Dyck words.

Let us call well-formed parenthesizations which are prefixed by an extra opening parenthesis Dyck-1 words (or Dyck words of depth 1). That is, a string \(w\) is a Dyck-1 word, or simply Dyck-1, iff \(d(w) = 1\) and \(d(\alpha) \geq 1\) for every non-empty prefix \(\alpha\) of \(w\).

We first prove two remarkable properties of rotations of Dyck-1 words:

**Lemma 2.1** There are no Dyck-1 words among the proper rotations of a Dyck-1 word.

**Proof.** Let \(w = \alpha \beta\) be a Dyck-1 word and \(u = \beta \alpha\) its proper rotation. Since \(w\) is Dyck-1, we have \(d(w) = 1\) and \(d(\alpha) \geq 1\). Since \(d(u) = d(\beta) + d(\alpha) = d(w)\), we have that \(d(\beta) \leq 0\). This means that \(u\) cannot be Dyck-1. \(\Box\)

A consequence of Lemma 2.1 is that a Dyck-1 word of length \(2n + 1\) has \(2^n + 1\) different rotations:

**Lemma 2.2** All rotations of a Dyck-1 word are different.

**Proof.** Let \(w = a_1a_2\ldots a_m\) be a Dyck-1 word of symbols \(a_1, a_2, \ldots, a_m \in \{(),\}\). Let \(r_i\) denote its \(i\)th rotation \(a_1a_{i+1}\ldots a_ma_1a_2\ldots a_{i-1}\), where \(1 \leq i \leq m\). Assume for the contrary that \(w\) has two rotations \(r_i\) and \(r_j\) with \(1 \leq i < j \leq m\) which are identical. Then also \(a_1\ldots a_{i-1} = a_{j-i+1}\ldots a_{j-1}\). This means that \(w\) equals its proper rotation \(r_{j-i+1}\), which is according to Lemma 2.1 not possible. \(\Box\)

The two lemmas above imply the next result, which says that rotations partition the set \(A_n\) in \(C_n\) equivalence classes of equal cardinality:

**Lemma 2.3** Every string of \(A_n\) has \(2n + 1\) different rotations, out of which exactly one is a Dyck-1 word.

**Proof.** Let \(w \in A_n\). It suffices to show that \(w\) has a rotation \(w'\) which is a Dyck-1 word. Because rotations of this \(w'\) are also rotations of \(w\), Lemma 2.1 gives that no other of them is Dyck-1, and Lemma 2.2 gives that all \(2n + 1\) of them are different.

If \(w\) is a Dyck-1 word, there is nothing further to prove. Otherwise let \(u\) be a non-empty prefix of \(w\) such that \(d(u)\) is minimal; if there are
several such prefixes, we take $u$ to be the longest of them. Let $v$ be the corresponding suffix of $w$, that is, $w = uv$. Since $d(w) = 1$, string $w$ must fail the Dyck-1 condition by $d(u) < 1$. Now $w' = vu$ is a rotation of $w$ with $d(w') = 1$, which satisfies also the other Dyck-1 condition that each of its non-empty prefixes has a positive depth: First, let $\alpha$ be a non-empty prefix of $w'$ which is also a prefix of $v$. Since $u$ was chosen to be the longest of the minimal-depth prefixes of $w$ and $u\alpha$ is a longer prefix of $w$, we have that $d(u\alpha) = d(u) + d(\alpha) \geq d(u) + 1$, which gives that $d(\alpha) \geq 1$. Second, let $\alpha$ be a prefix of $w'$ which contains also some characters of $u$, that is, $\alpha = vu'$ for some non-empty prefix $u'$ of $u$. Since $u$ is a minimal-depth prefix of $w$, we have $d(u') \geq d(u)$ and thus $d(\alpha) = d(vu') = d(v) + d(u') \geq d(v) + d(u) = 1$.

Since there are $C_n$ different Dyck-1 words of length $2n + 1$, we have the following corollary:

**Corollary 2.4** The set $A_n$ consists of $C_n$ equivalence classes under rotation, and each of them has $2n + 1$ members.

According to the Corollary $|A_n| = (2n + 1)C_n$, and thus

$$C_n = \frac{|A_n|}{2n + 1} = \left(\frac{2n + 1}{2n + 1}\right) = \frac{(2n + 1)2n \cdots (n + 2)}{(2n + 1)n!} = \frac{2n(2n - 1) \cdots (n + 1)}{n!(n + 1)} = \frac{1}{n + 1} \binom{2n}{n}$$

### 3 Concluding remarks

Our argument is closely related to a lemma which states that a sequence of integers which add up to +1 has exactly one rotation such that all of its partial sums are positive. Graham, Knuth and Patashnik (1990, p. 345) attribute this result to Raney (1960). Indeed, this “Raney’s Lemma” can be proved by the same argument as Lemmas 2.3 and 2.1, simply by interpreting $d(w)$ as “sum of a subsequence” instead of “depth of a sub-word”. Graham, Knuth and Patashnik (1990, p. 346) give for “Raney’s Lemma” a geometric proof, which is debatably more involved than our simple combinatorial argument.
References


