Issue Date: 21.06.2013
Version: 2013-06-21 13:30

Exercise Sheet 9 for Combinatorial Algorithms, SS 13

Hand In: *Until Monday, 01.07.2013, 12:00,*
box in the group's hallway or email to `wild@cs.uni....`

Problem 15

2 + 2 + 1 points

Consider the labeled combinatorial class $\mathcal{Q}$ implicitly defined by the specification

$$\Theta\mathcal{Q} \;=\; \mathcal{Q} \star \mathcal{Z} \star \mathcal{Q}\,, \tag{1}$$

where Θ is the *pointing operator* as defined by Flajolet et al. [FZV94].

a) Write down the functional equation for the (exponential) generating function $Q(z)$ for $\mathcal{Q}$ according to the specification (1). Translate this equation of generating functions to an equation of the coefficients, thereby deriving a recurrence equation for Q_n, the number of objects of size n in $\mathcal{Q}$.

b) Note that (1) does not specify Q_0. Can you prove a simple closed form of Q_n assuming $Q_0 = 1$? For an educated guess, you might compute the first few entries, say $Q_0, \ldots, Q_6$, and ask the *On-Line Encyclopedia of Integer Sequences* for help — but don't forget to prove your claims.

c) Show that $\mathcal{Q}$ is isomorphic to a well-known combinatorial class.

Problem 16

1 + 4 + 3 points

In this exercise we consider the random generation of RNA secondary structures. RNA is an important single-stranded relative of DNA, which exhibits rich folding structures. For details, refer to our lecture *Computational Biology II*.

Here, it suffices to know that secondary structures are isomorphic to *Motzkin words*, which are words over $\Sigma = \{\texttt{(}, \texttt{)}, \texttt{*}\}$, where `(` and `)` form well-nested pairs and `*` can appear anywhere, e. g., "`*((**(***))*((**))`" is a Motzkin word, but neither "`)(`" nor "`(())(`" are.

Formally, we define the *unlabeled* combinatorial class $\mathcal{S}$ of RNA secondary structures as

$$\mathcal{S} \;=\; \epsilon \;+\; \mathcal{Z}_{*} \times \mathcal{S} \;+\; \mathcal{Z}_{(} \times \mathcal{S} \times \mathcal{Z}_{)} \times \mathcal{S}\,. \tag{2}$$

a) Note that Flajolet et al. only consider labeled classes, whereas the secondary structures defined above are unlabeled. Argue why their approach of random generation is also applicable to the unlabeled specification (2).

b) Use the method of described by Flajolet et al. [FZV94] to design an algorithm to generate a random secondary structure of given size n chosen uniformly among all structures $s \in \mathcal{S}_n$ of size n. Explicitly write down at least the following intermediate steps:

- the *standard specification* for $\mathcal{S}$,
- recurrence equations for the number of objects of all classes that occur in the standard specification and
- the generation procedures for all those classes.

You may directly simplify the generating procedures where the special case makes it possible, but explain your simplifications.

c) Implement the random generation procedure in a programming language of your choice. Draw 1000 random structures of size $n = 100$ and draw a histogram of the number of unpaired bases, i. e. the number of `*` atoms. Do you have a guess for the distribution?

References

[FZV94] Philippe Flajolet, Paul Zimmerman, and Bernard Van Cutsem. A calculus for the random generation of labelled combinatorial structures. *Theoretical Computer Science*, 132(1-2):1–35, September 1994. `doi:10.1016/0304-3975(94)90226-7`.