

MATH 343, SPRING 2012, ASSIGNMENT 4

DUE THURSDAY FEBRUARY 28, 2013 IN CLASS

Do **any three** of the following four problems. If you do more than three, only the first three will be graded.

- (1)
 - (a) Implement an unrank for k -subsets (either the one we discussed in class or one using the corank trick) in a language of your choice. Use this to randomly generate k -subsets of n for various n . Time your program and plot the timings as a function of n .
 - (b) Using `combstruct`'s draw function, generate random k -subsets of n for various n . Time this and plot the timings as a function of n . The maple command `time` may be helpful to you.
 - (c) Compare the results of the two previous parts. Do you think you are using the same algorithm that `combstruct` does?
- (2) A Dyck prefix is a path beginning at $(0, 0)$, using the steps $(1, 1)$ and $(1, -1)$, and which does not go strictly below the x -axis. There is no further restriction on where it ends. The length of a Dyck prefix is the number of steps.
 - (a) A Dyck prefix can be decomposed as a largest possible Dyck path followed by one or more $(1, 1)$ steps and then another Dyck path and so on. Using this idea find a specification for the combinatorial class of Dyck prefixes.
 - (b) Give a specification for the combinatorial class of Dyck prefixes with final step ending at height y . *Note that this is essentially the same class we counted in a very different way in lecture when we were ranking Dyck paths.*
- (3) Write pseudocode for the Prüfer encoding which runs in $O(n \log n)$ where n is the size of the tree. You may choose whatever representation you like for the tree. You can assume that you have available a data structure which can insert and remove elements in $O(\log n)$ and can find the maximum element in $O(1)$ (i.e. a priority queue). Justify that your algorithm runs in $O(n \log n)$.
- (4) A multiset is like a set but can have repeated elements (order still doesn't matter). A k -multiset is a multiset of size k . Give lexicographic ranking and unranking algorithms for k -multisets of $\{1, \dots, n\}$.