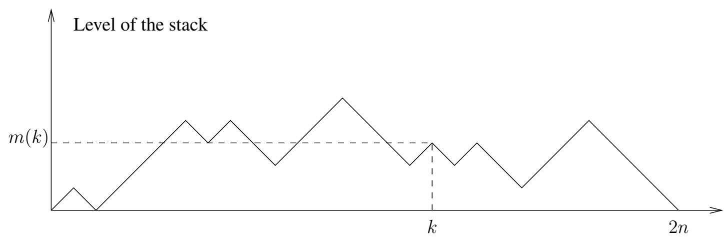
Master of Science in Informatics at Grenoble

UE Mathematics for Computer Science

Homework, October 2010

Counting stacks *(10 points)*

Consider a stack with the two primitives *push* and *pop*. An execution of a program consists in n operations *push* and n *pop* which could be interleaved. The execution is represented by a *mountain*, the function $m(k)$ that gives the level of the stack after k operations.



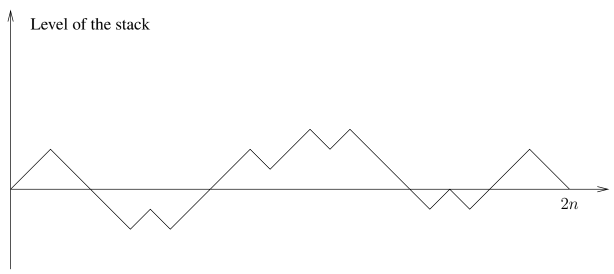
Denote by M_n the number of *mountains* with n *push* (up-stroke) and n *pop* (down-stroke) operations and set $M_0 = 1$.
Question 1.1 : Small n cases

For $n = 1, 2, 3$ give the possible *mountains* and deduce M_1, M_2, M_3.

Question 1.2 : Graph formulation

Propose a formulation of the problem as a path problem in a graph.

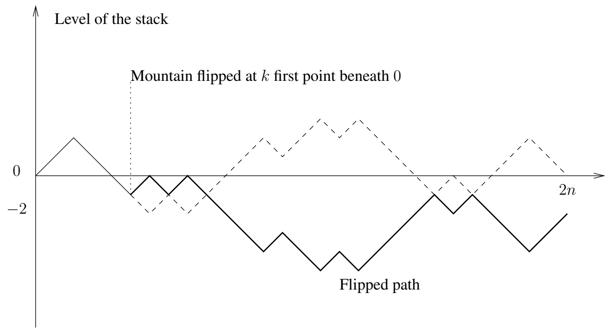
An extended *mountain* of length $2n$ with n up-strockes and n down-strockes allows to be under the sea level (bad mountains):



Question 1.3 : Extended *mountains*

Compute the number of extended mountains with length $2n$.

The flip operation consists in exchanging all the slopes after the first passage below 0:



Question 1.4 : Flipped mountains

Show that the set of bad *mountains* is in bijection with the set of *mountains* with $n-1$ up-strokes and $n+1$ down-strokes.

Question 1.5 : Computation

Prove that

$$M_n = \frac{1}{n+1}\binom{2n}{n} = \frac{2n!}{(n+1)!n!}. \tag{1}$$

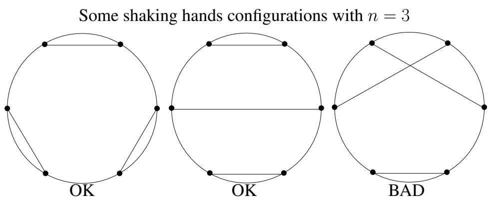
Question 1.6 : Recurrence relation

Show directly on mountain diagrams that the M_n numbers satisfy the recurrence equation:

$$M_n = M_0 M_{n-1} + M_1 M_{n-2} + \cdots + M_{n-1} M_0. \tag{2}$$

Question 1.7 : Random mountains

From a uniform random generator *random()* propose an algorithm that generates mountains uniformly.



Question 1.8 : Shake hands

Suppose that $2n$ persons are seated around a table, how many ways could they shake hands without crossing ?

Question 1.9 : Generating function

From the recurrence equation 2 compute the generating function of the sequence and deduce formula 1.