Tobias Friedrich, Timo Kötzing, Andrew Sutton **Summer Term 2015**

Homework 1 – "Heuristic Optimization"

https://hpi.de/friedrich/teaching/ss15/heuristic-optimization.html

First we want to make some formalities more clear.

For **projects:** The projects you can work on in groups of 3-4 people. Submissions are via email to `Clemens.Frahnow@student.hpi.uni-potsdam.de`.

For **homeworks:** The homeworks train also your skills for writing down math and for reasoning mathematically; thus, we require individual submissions. You may discuss the problems with others, but we strongly encourage you to give it a try by yourself first. Your submissions need to be on paper (with your name on it and stapled, if you have multiple sheets). You submit homeworks by putting them into the box labeled *homeworks* on the first floor of building A; if you cannot find such a box, slip them under the door of office A-1.12.

This homework is due **Wednesday, April 29, 12:45**. In this homework we practice a bit reasoning about probabilities.

Exercise 1 *A randomized algorithm A is known to have a very erratic run time, sometimes its slow, sometimes very fast. We let T_n be the random variable describing the run time of A on inputs of size n. Suppose that with some math we were able to show that $E(T_n) \leq 5n^2$.*

(a) Show that $P(T_n \geq n^3) = O(1/n)$.

(b) Give an example for T_n such that $P(T_n \geq n^3) = \Theta(1/n)$.

We don't like this comparatively high chance of having a run time of more that n^3. Thus, we now want to design an algorithm A' which gets the result with only a very small chance of taking very long. We use the following idea: We let A run for a time of $t_0 = 25n^2$. If it was not successful so far, we restart it on the same input again for a time of t_0 and so on, until at some point we will be done within time t_0. We let T'_n be the random variable describing the run time of A' on inputs of size n.

(c) Show that $P(T'_n \geq n^3) = 2^{-\Omega(n)}$.