1. Ski Rental. You want to start skiing but you do not own skis. Every day, you can decide to either rent the skis for that day for \$1 or buy them for \$$C$ and own them forever. However, you have a premonition that there is an adversary who will one day break your legs and end your skiing career. The problem is that you do not know when they are going to do so, i.e. on the N^{th} day, the adversary will break your legs but you do not know N in advance. What is the most efficient way to spend your money?

   Example. $C = 10, N = 2$. If we buy the skis on the first day, we get one use of them, thus paying \$10 for one skiing session. Compare this to only renting them on the first day, then we pay \$1 for one skiing session.

   Tasks.

   a) Design a deterministic 2-competitive algorithm.

   b) Show that it is optimal.

   c) Consider the following randomized algorithm: we uniformly randomly independently (of C) choose $p \in (0, 1)$ and on each day we decide with probability p to buy the skis and with probability $1 - p$ to rent the skis. How competitive is this algorithm?

   d) Can we improve the algorithm in the previous part? By, e.g. choosing better probabilities?

2. Cow Path. A two-dimensional cow wants to escape a fenced area. The exit is at the either end of the fenced area, but the cow does not know which one is the exit. Furthermore, it is very foggy and the cow does not see anything in front of it. It will only recognize the exit by stepping on it.

   We can without loss of generality assume that we start at 0 and the left end is at $\ell \in \mathbb{Z}^-$ or at $r \in \mathbb{Z}^+$. Either one of these points is the exit. The cow moves at unit speed.

   Tasks.

   a) Design a deterministic strategy in which the cow moves as little as possible.

   b) Can we do better with randomization?