EXERCISE 1 Assume that you are given a randomized primitive BIAS(), returning 1 with probability p, and 0 with probability $1-p$, independently at each call. Assume that p is not known. Design an algorithm UNBIAS() that calls BIAS() repeatedly and returns 0/1 with probability 1/2 (clearly, UNBIAS cannot use RANDOM({0,1})). Analyze the number of calls to BIAS() needed as a function of the unknown parameter p

ADDITIONAL EXERCISES:

1. Implement RANDOM($\{0,1,2\}$) using BIAS()
2. Implement RANDOM($\{0,\ldots,n-1\}$) using RANDOM($\{0,1\}$) (n ARBITRARY)

EXERCISE 2 (Partial coupon collecting)

Given a constant $c > 1$ determine an upper bound $m_c(n)$ to the number of calls to RANDOM($\{1, \dots, n\}$) so that the expected number of distinct values returned is at least $\frac{n}{c}$