## Randomized algorithms

1. Consider the following randomized algorithm for generating biased random bits. The subroutine FAIR-COIN returns either 0 or 1 with equal probability; the random bits returned by FAIRCOIN are mutually independent.

```
ONEINTHREE:
```

```
if FAIRCOIN = 0
return 0
else
return 1- ONEINTHREE
```

- (a) Prove that ONEINTHREE returns 1 with probability 1/3.
- (b) What is the exact expected number of times that this algorithm calls FAIRCOIN?
- (c) Now suppose you are given a subroutine ONEINTHREE that generates a random bit that is equal to 1 with probability 1/3. Describe a FAIRCOIN algorithm that returns either 0 or 1 with equal probability, using ONEINTHREE as your only source of randomness.
- 2. (a) Suppose you have access to a function FAIRCOIN that returns a single random bit, chosen uniformly and independently from the set  $\{0, 1\}$ , in O(1) time. Describe and analyze an algorithm RANDOM(n), which returns an integer chosen uniformly and independently at random from the set  $\{1, 2, ..., n\}$ .
  - (b) Suppose you have access to a function FAIRCOINS(k) that returns k random bits (or equivalently, a random integer chosen uniformly and independently from the set  $\{0, 1, ..., 2^k 1\}$ ) in O(1) time, given any non-negative integer k as input. Describe and analyze an algorithm RANDOM(n), which returns an integer chosen uniformly and independently at random from the set  $\{1, 2, ..., n\}$ .
- 3. Suppose you are given a graph G with weighted edges, and your goal is to find a cut whose total weight (not just number of edges) is smallest.
  - (a) Describe an algorithm to select a random edge of G, where the probability of choosing edge e is proportional to the weight of e.
  - (b) Prove that if you use the algorithm from part (a), instead of choosing edges uniformly at random, the probability that the single-phase randomized min cut algorithm (GUESSMINCUT, in Jeff Erickson's notes) returns a minimum-weight cut is still  $\Omega(1/n^2)$ .
  - (c) What is the running time of your modified GUESSMINCUT algorithm?
- 4. Prove that the single-phase randomized min cut algorithm (GUESSMINCUT, in Jeff Erickson's notes) returns the second smallest cut in its input graph with probability  $\Omega(1/n^3)$ . (The second smallest cut could be significantly larger than the minimum cut.)
- 5. Consider the following randomized algorithm for choosing the largest bolt. Draw a bolt uniformly at random from the set of n bolts, and draw a nut uniformly at random from the set of n nuts. If the bolt is smaller than the nut, discard the bolt, draw a new bolt uniformly at random from the unchosen bolts, and repeat. Otherwise, discard the nut, draw a new nut uniformly at random from the unchosen nuts, and repeat. Stop either when every nut has been discarded, or every bolt except the one in your hand has been discarded.

What is the expected number of nut-bolt tests performed by this algorithm? Prove your answer is correct. [Hint: What is the expected number of unchosen nuts and bolts when the algorithm terminates?]

6. The IRS receives, every year, n forms with personal tax returns. The IRS, of course, can not verify all n forms, but they can check some of them. Describe an algorithm, as fast as possible, that decides whether the number of incorrect tax forms is larger than  $\epsilon n$ , where  $\epsilon$  is a prespecified constant between 0 and 1.

The decision of the algorithm is considered to be incorrect if it declares that the number of incorrect forms is smaller than  $\epsilon n$ , but it is in fact larger than  $2\epsilon n$ . Similarly, the algorithm is considered to be incorrect if it claims that the number of incorrect forms is larger than  $2\epsilon n$ , where it is in fact smaller than  $\epsilon n$ . (Namely, if the number of incorrect forms is between  $\epsilon n$  and  $2\epsilon n$ , any of the two answers are acceptable.)

Your algorithm should output a correct result with probability  $\geq 1 - 1/n^{10}$ . What is the running time of your algorithm, assuming that verifying the correctness of a single tax form takes O(1) time? (Hint: Use the Chernoff inequalities.)