

**Write your answers in the separate answer booklet.**

Please return this question handout and your cheat sheets with your answers.

---

1. Let  $G = (V, E)$  be an arbitrary undirected graph. A **triple-Hamiltonian circuit** in  $G$  is a closed walk in  $G$  that visits every vertex of  $G$  exactly *three* times. **Prove** that it is NP-hard to determine whether a given undirected graph has a triple-Hamiltonian circuit. [Hint: Modify your reduction for double-Hamiltonian circuits from Homework 10.]

2. Marie-Joseph Paul Yves Roch Gilbert du Motier, Marquis de Lafayette, colonial America's favorite fighting Frenchman, needs to choose a subset of his ragtag volunteer army of  $m$  soldiers to complete a set of  $n$  important tasks, like "go to France for more funds" or "come back with more guns". Each task requires a specific set of skills, such as "knows what to do in a trench" or "ingenuitive and fluent in French". For each task, exactly  $k$  soldiers are qualified to complete that task.

Unfortunately, Lafayette's soldiers are extremely lazy. For each task, if Lafayette chooses more than one soldier qualified for that task, each of them will assume that someone else will take on that task, and so the task will never be completed. A task will be completed if and only if exactly one of the chosen soldiers has the necessary skills for that task.

So Lafayette needs to choose a subset  $S$  of soldiers that maximizes the number of tasks for which *exactly one* soldier in  $S$  is qualified. Not surprisingly, Lafayette's problem is NP-hard.

- (a) Suppose Lafayette chooses each soldier independently with probability  $p$ . What is the *exact* expected number of tasks that will be completed, in terms of  $p$  and  $k$ ?
  - (b) What value of  $p$  maximizes this expected value?
  - (c) Describe a randomized polynomial-time  $O(1)$ -approximation algorithm for Lafayette's problem. What is the expected approximation ratio for your algorithm?
3. Suppose we are given a set of  $n$  rectangular boxes, each specified by their height, width, and depth in centimeters. All three dimensions of each box lie strictly between 10cm and 20cm, and all  $3n$  dimensions are distinct. As you might expect, one box can be nested inside another if the first box can be rotated so that it is smaller in every dimension than the second box. Boxes can be nested recursively, but two boxes cannot be nested side-by-side inside a third box. A box is *visible* if it is not nested inside another box.

Describe and analyze an algorithm to nest the boxes, so that the number of visible boxes is as small as possible.

4. Hercules Mulligan, a tailor spyin' on the British government, has determined a set of routes and towns that the British army plans to use to move their troops from Charleston, South Carolina to Yorktown, Virginia. (He took their measurements, information, and then he smuggled it.) The American revolutionary army wants to set up ambush points in some of these towns, so that every unit of the British army will face at least one ambush before reaching Yorktown. On the other hand, General Washington wants to leave as many troops available as possible to help defend Yorktown when the British army inevitably arrives.

Describe an efficient algorithm that computes the smallest number of towns where the revolutionary army should set up ambush points. The input to your algorithm is Mulligan's graph of towns (vertices) and routes (edges), with Charleston and Yorktown clearly marked.

5. Consider the following randomized algorithm to approximate the smallest vertex cover in an undirected graph  $G = (V, E)$ . For each vertex  $v \in V$ , define the *priority* of  $v$  to be a real number between 0 and 1, chosen independently and uniformly at random. Finally, let  $S$  be the subset of vertices with higher priority than at least one of their neighbors:

$$S := \left\{ v \in V \mid \text{priority}(v) > \min_{uv \in E} \text{priority}(u) \right\}$$

- (a) What is the probability that the set  $S$  is a vertex cover of  $G$ ? **Prove** your answer is correct. (Your proof should be *short*.)
- (b) Suppose the input graph  $G$  is a cycle of length  $n$ . What is the *exact* expected size of  $S$ ?
- (c) Suppose the input graph  $G$  is a **star**: a tree with one vertex of degree  $n - 1$  and  $n - 1$  vertices of degree 1. What is the *exact* probability that  $S$  is the *smallest* vertex cover of  $G$ ?
- (d) Again, suppose  $G$  is a star. Suppose we run the randomized algorithm  $N$  times, generating a sequence of subsets  $S_1, S_2, \dots, S_N$ . How large must  $N$  be to guarantee with high probability that some  $S_i$  is the minimum vertex cover of  $G$ ?
6. After the Revolutionary War, Alexander Hamilton's biggest rival as a lawyer was Aaron Burr. (Sir!) In fact, the two worked next door to each other. Unlike Hamilton, Burr cannot work non-stop; every case he tries exhausts him. The bigger the case, the longer he must rest before he is well enough to take the next case. (Of course, he is willing to wait for it.) If a case arrives while Burr is resting, Hamilton snatches it up instead.

Burr has been asked to consider a sequence of  $n$  upcoming cases. He quickly computes two arrays  $\text{profit}[1..n]$  and  $\text{skip}[1..n]$ , where for each index  $i$ ,

- $\text{profit}[i]$  is the amount of money Burr would make by taking the  $i$ th case, and
- $\text{skip}[i]$  is the number of consecutive cases Burr must skip if he accepts the  $i$ th case. That is, if Burr accepts the  $i$ th case, he cannot accept cases  $i + 1$  through  $i + \text{skip}[i]$ .

Design and analyze an algorithm that determines the maximum total profit Burr can secure from these  $n$  cases, using his two arrays as input.