Minitest 5 – Solutions

1. If a probabilistic algorithm provides a correct YES/NO answer with probability at least 2/3, and we run it independently n times, the probability that it gives the wrong answer more than n/2 times is smaller than exp(-0.001n).

Answer: TRUE

Explanation. We can apply the **Chernoff Bound** [Satz 2.70. in Script / on Cheatsheet]:

$$Pr[X \leq (1-\delta)\mathbb{E}[X]] \leq e^{-\frac{1}{2}\delta^2\mathbb{E}[X]}$$

In this case, let X be the number of correct answers. With $p \geq 2/3$, the expectation $\mathbb{E}[X] \geq \frac{2}{3}n$. We consider an error to occur if $X \leq n/2$. Setting $(1-\delta)\frac{2}{3}n = \frac{1}{2}n$ gives $\delta = 1/4$.

$$Pr[X \leq n/2] \leq e^{-\frac{1}{2}(\frac{1}{4})^2(\frac{2}{3}n)} = e^{-\frac{1}{48}n}$$

Since $1/48 \approx 0.02 > 0.001$, $e^{-n/48} < e^{-0.001n}$.

2. There is a probabilistic algorithm testing if n is a prime number in time polynomial in log n.

Answer: TRUE

Explanation. The **Miller-Rabin Primality Test** [Chapter 2.8.3 in Script].

3. Let A be a probabilistic algorithm whose output is always either 0 or 1, and suppose $E[A] = s > 0$. Let $0 < \epsilon < 1$ and $0 < \delta < 0.5$. If we run A independently $m = \lceil 100 \log(1/\delta)/(s\epsilon^2) \rceil$ times and define ś as the average of the outputs, then with probability at least $1 - \delta$: ś $\in [(1-\epsilon)s, (1+\epsilon)s]$.

Answer: TRUE

Explanation. This is connected to the **Target Shooting** principle found in the Script and Cheatsheet. By repeating the experiment $m \approx s^{-1}\epsilon^{-2} \ln(\delta^{-1})$ times, the Chernoff bound guarantees that the average ś stays within an ϵ-relative error of the expectation s with probability $\geq 1 - \delta$.

4. We can transform every Las Vegas algorithm with known expected running time at most T into a randomized algorithm whose running time is at most 10T and whose success probability is at least 0.9.

Answer: TRUE

Explanation. This identity follows directly from **Markov's Inequality** [Satz 2.67. / Cheatsheet]: $Pr[X \geq t \cdot \mathbb{E}[X]] \leq 1/t$. For a time limit $t = 10T$, the probability of failing to finish is $Pr[X \geq 10T] \leq T/10T = 0.1$, yielding a success probability of ≥ 0.9.

5. Every Monte Carlo algorithm can be converted into a Las Vegas algorithm.

Answer: FALSE

Explanation. A conversion is only guaranteed if the output is **efficiently verifiable**. If there is no efficient check for correctness, we cannot turn the probabilistic result of a Monte Carlo algorithm into the guaranteed result of a Las Vegas algorithm.

6. Sarah has two special coins: one always lands heads, and the other always lands tails. She picks one of the two coins uniformly at random and flips it n = 1000 times. Let X be the number of heads observed. Then, for $\delta := 1/4$: $\Pr[X \geq (1+\delta)n/2] \leq e^{-\delta^2 n/6}$.

Answer: FALSE

Explanation. The **Chernoff Bound** explicitly requires the random variables $X_1, \ldots, X_n$ to be **independent**. Here, the flips are perfectly correlated once the coin is chosen (either all heads or all tails), so the bound does not apply. $\Pr[X \geq (1+\delta)n/2] = 1/2$

7. A deterministic algorithm can always be viewed as a randomized algorithm.

Answer: TRUE

Explanation. A deterministic algorithm is a special case of a randomized algorithm where no random bits are used, or where the "random" choice happens with probability 1.

8. Let X, Y, Z be three random variables such that X, Y, Z are independent. Then it always holds that E[X + Y · Z] = E[X] + E[Y]·E[Z].

Answer: TRUE

Explanation. This follows from Linearity of Expectation ($\mathbb{E}[X + YZ] = \mathbb{E}[X] + \mathbb{E}[YZ]$) and the Multiplicativity rule for independent variables Y and Z ($\mathbb{E}[YZ] = \mathbb{E}[Y]\mathbb{E}[Z]$), both of which can be found on the Cheatsheet (and in the Script ofc).

9. The expected running time of Quickselect is O(n).

Answer: TRUE

Explanation. Chapter 2.8.2 in Script

10. Quicksort is a Monte Carlo algorithm.

Answer: FALSE

Explanation. Quicksort is a **Las Vegas Algorithm**. Its result is always correctly sorted, and only the **running time** is a random variable based on pivot selection.