CS 5200 Homework - 2

Instructor Avah Banerjee
Due: September 30, 11:59PM

Problem 1 Given an integer a and a positive number N, compute the multiplicative inverse of a modulo N.

1. Write the algorithm for computing the multiplicative inverse.
2. Implement the algorithm using Python.
3. Analyze the running time of your algorithm.
4. Test the performance (determine running time plot by counting number of atomic operations in your code) of your algorithm empirically.

Problem 2 Recall the `fastmultiply(x, y)` function which multiplies two n-bit binary integers x and y in time n^a, where $a = \log_2 3$. Convert the decimal integer 10^n (a 1 followed by n zeros) into binary.

Algorithm - 1

```
function pwr2bin(n):
    if n = 1:
        return 1010_2
    else:
        z = ???
        return fastmultiply(z, z)
```

1. Fill in any missing details in the algorithm.
2. Establish and solve a recurrence relation for the running time of the algorithm.

Problem 3 Given a random $n \times n$ matrix, M, whose entries are i.i.d with either 0 or 1 with equal probability, and a random Boolean vector, v, of length n whose entries also i.i.d and have equal probability of being 0 or 1, what is the probability that, when v is multiplied with M, the result is the 0 vector?

Note: In this context, multiplications refer to AND operations, and additions are OR operations. That is the j^{th}-entry of $u = Mv$ is $u_j = \bigvee_i (M_{ik} \bigwedge v_k)$.

1. Derive the probability expression or formula for the scenario described above.

2. Provide a clear and concise explanation for your derived expression or formula.

3. Implement a simulation in Python to empirically verify your result.

4. Compare the theoretical probability with the empirical result from your simulation. Discuss any observed differences or similarities and their potential causes.

Submission Guidelines:

- Your solution should be neatly formatted, and each step of your reasoning should be explained clearly.
- For the Python implementation, ensure your code is clean, well-commented, and runs without errors.
- When analyzing running times, be precise and justify your claims.
- Empirical tests should be conducted on a variety of test cases, and the results should be discussed in the context of the theoretical analysis.