

Write-Up for the Hiring Problem Simulation

1. Description of the Code

The provided Java and Python program simulates a variant of the hiring problem. In this version of the hiring problem, each candidate is assigned a suitability rating that is an integer chosen uniformly at random from 0 to (x) , where (x) is defined differently under three cases:

- **Case A:** $(x = \lceil \log_2 n \rceil)$
- **Case B:** $(x = n)$
- **Case C:** $(x = n^3)$

Here, (n) represents the number of candidates, and we simulate the process for every (n) from 1 to 1000. For each (n) and each case, the program runs 10 independent simulation trials. During a trial, the program “hires” a candidate if the candidate’s rating is strictly greater than the current best rating.

The simulation for each (n) involves the following steps:

1. Determine the Range of Ratings $((x))$:

- For each candidate count (n) , the value of (x) is computed for each case (A, B, and C). For example, in Case A, $(x = \lceil \log_2 n \rceil)$.

2. Simulation of the Hiring Process:

- The simulation starts with no candidate hired (using a `currentBest` initialized to -1).
- For each candidate, a random rating is generated using Java’s `Random` class.
- If the candidate’s rating is strictly greater than the `currentBest`, the candidate is hired, and `currentBest` is updated to that rating.
- The number of hires is recorded for that trial.

3. Computation of Statistics:

- After running the required number of trials (10 in this case), the program computes:
 - The **mean (average)** number of hires.
 - The **standard deviation** of the hires across the trials.

4. Output to CSV:

- The code writes the output to a CSV file named `results.csv`.
- Each row in the CSV file corresponds to a particular value of (n) and contains the following columns:
 - (n) (number of candidates)
 - For each case (A, B, and C): the value of (x) , the computed mean number of hires, and the standard deviation.
- This output is later imported into Python’s `matplotlib` for plotting and further analysis.

2. Analysis

The simulation is designed to study the behavior of the hiring problem under different conditions of the candidate rating range:

1. Case A ($x = \lceil \log_2 n \rceil$):

- Since x grows very slowly with n (logarithmically), the range of possible ratings is small.
- This means that after a candidate with a high rating is hired, it becomes very unlikely to find a candidate with a strictly higher rating.
- **Expected Behavior:** The number of hires will be relatively low and will tend to plateau as n increases.

2. Case B ($x = n$):

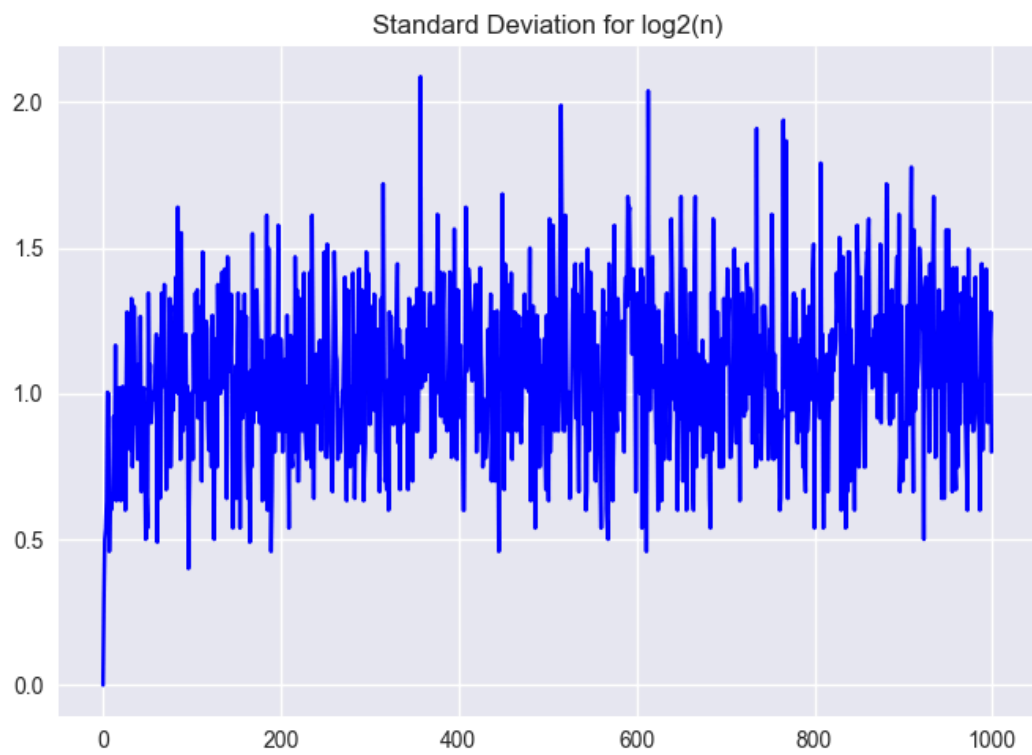
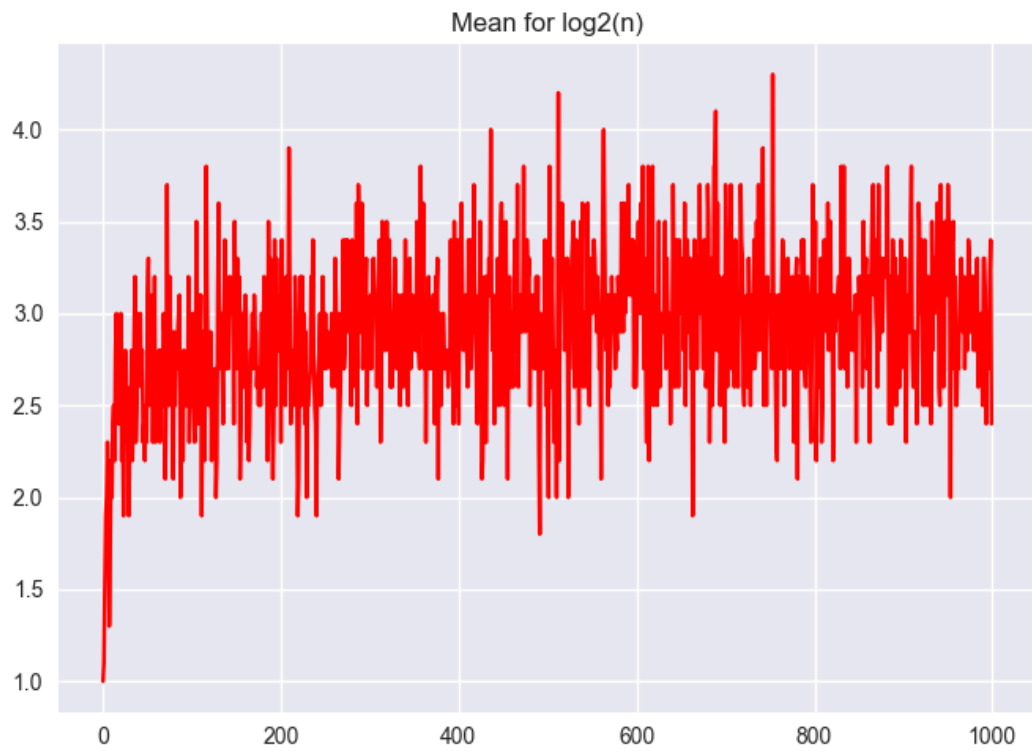
- Here, the range of possible ratings increases linearly with the number of candidates.
- This makes ties less frequent, and the process starts to resemble the classical hiring problem.
- **Expected Behavior:** The expected number of hires should be similar to the harmonic series, approximately $\ln n$. This means that as n increases, the number of hires grows logarithmically.

3. Case C ($x = n^3$):

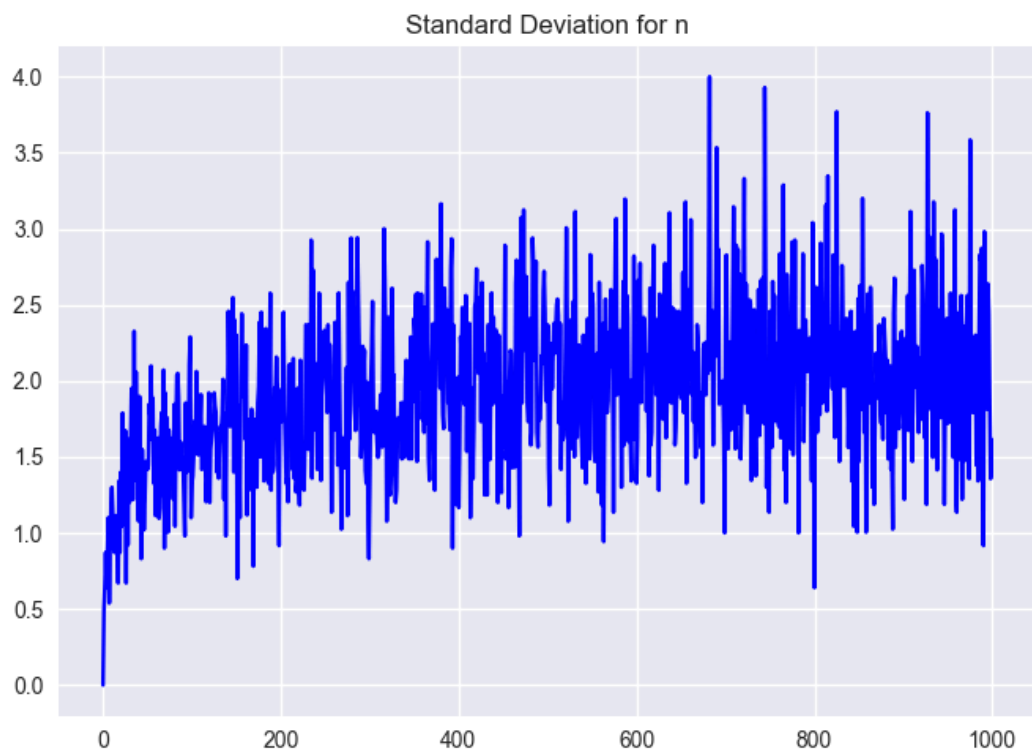
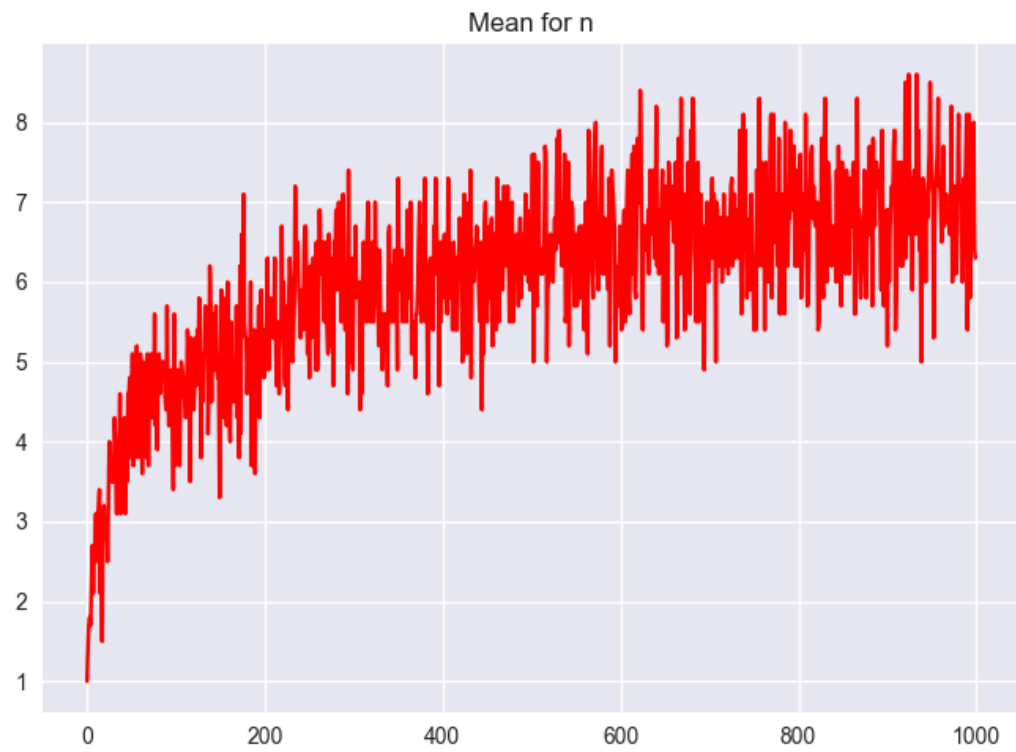
- With $x = n^3$, the range is very large compared to the number of candidates.
- Ties are highly unlikely, and the behavior should closely match the continuous case.
- **Expected Behavior:** The number of hires is expected to be nearly identical to the classical hiring problem, again following roughly $\ln n + O(1)$.

3. Plots

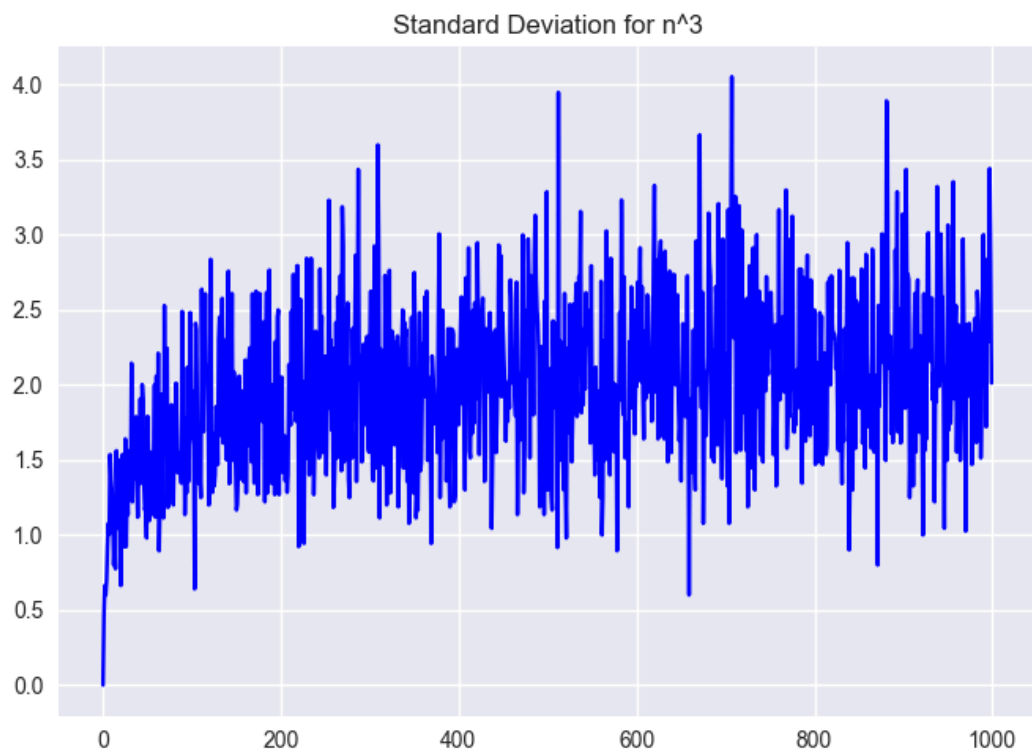
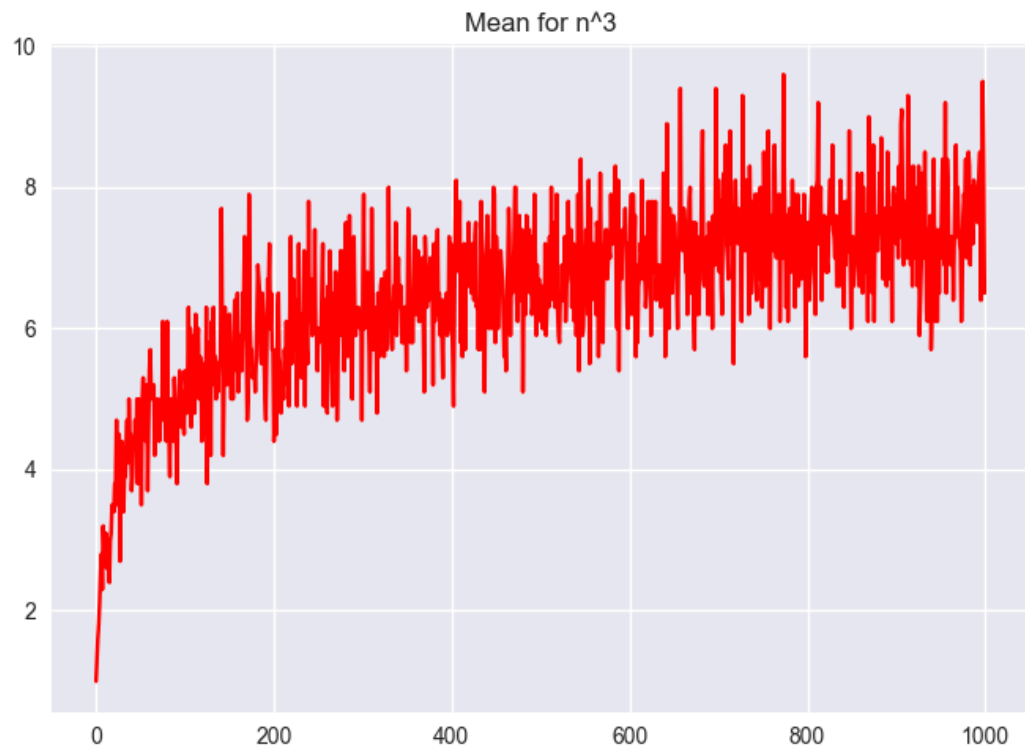
1. Mean and Standard Deviation graph for $x = \lceil \log_2 n \rceil$:



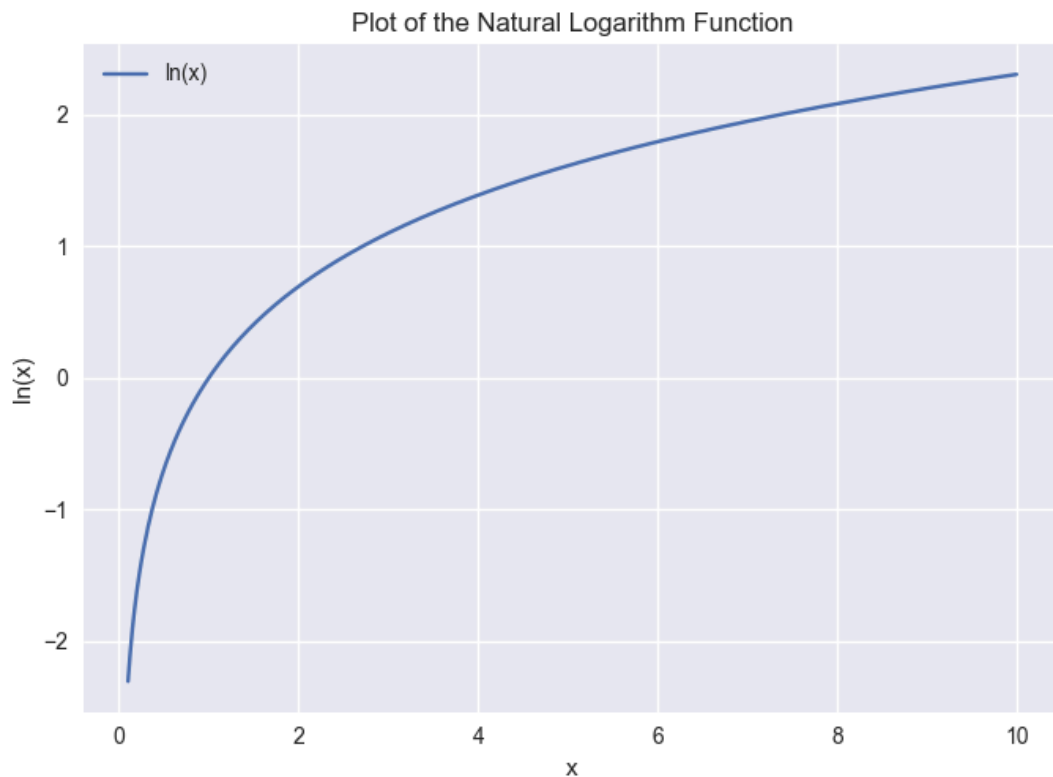
2. Mean and Standard Deviation graph for $x = n$:



3. Mean and Standard Deviation graph for $x = n^3$:



4. Plot of $\log(n)$ for comparison:



4. Conclusions

From the simulation and subsequent analysis, we can conclude the following:

- **Impact of the Rating Range (x) on Hiring:**
 - When the rating range is small (Case A), the number of hires is limited. This is because once the maximum possible rating is achieved, no further candidates can be hired. The simulation shows that the hiring process saturates quickly.
 - When (x) increases (Cases B and C), ties become rare, and the process more closely resembles the classical hiring problem. The number of hires grows in a logarithmic fashion with the number of candidates.

In summary, the simulation demonstrates that the discrete nature of candidate ratings and the choice of (x) significantly influence the number of hires. For large ranges of ratings, the hiring process follows the logarithmic behavior predicted by theory, while small ranges quickly lead to saturation in the number of hires. This insight is valuable for understanding both randomized algorithms and practical scenarios where candidate selection is based on discrete ratings.