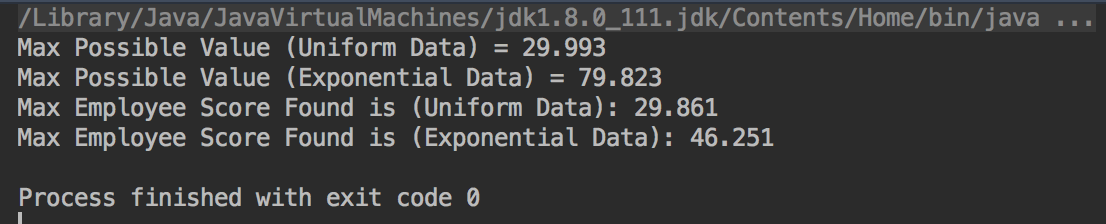
CMPE 365 Algorithms Lab 1

Robert Saunders

10194030

Please evaluate and run the code with the correct datasets, you will see the desired results in the console, see below for sample out using the es1.csv and us1.csv files:



As you can see the from the data the uniform data is much more accurate. The implementation of the code has a method called ‘getLargestNumber’ and it traverses the entire array to find the number with the largest magnitude, obviously this will always result in the best employee but is very costly, this was used to compare the best employee algorithm with the actual best. The alternative is the ‘findBestEmployee’ method that implements the best employee algorithm that we learned in class. In that method, we create a sample size of employees by taking the number of employees and dividing it by Euler’s number, this is historically a good sample size. We then find the best score in the sample size which becomes our baseline when comparing with the rest of the employees (dataset), the first employee to beat our baseline is considered the best employee. This method may not always result in the best employee but is much less involved and you will still result in a good employee.

The reason the uniform data is more accurate is because the employees are given in a random order, so the sample size is not diluted with good employees, it is a random group of skills. Whereas with the exponential data the sample size could consist of only good candidates or bad candidates because the data could be front loaded or back loaded, thus when comparing the rest of the dataset with the best employee would not be found and the last element in the dataset will be taken or alternatively the baseline could be very low and the first element in the remaining dataset would be taken, therefore at best you will get a mediocre employee.