* Name: Renee Alsop
* Date: 08/11/2017
* ID: P303319

**AT1.8 Testing**

**Test Procedure**

4 search algorithms were executed on the same randomly generated set of numbers in an array of 10000 integers to search for the same target number. Each search function outputted the time in nanoseconds that it took each algorithm to find the target number. This process was repeated 100 times, and the results written to a text file. This will illustrate the efficiency or lack thereof, of each search algorithm.

**So what are these "algorithms" you mention?**

**Built-in Binary Search**

The MSDN website does not elaborate on how this search functions, other than to say that it "Searches a range of elements in a one-dimensional sorted array for a value, using the [IComparable](https://msdn.microsoft.com/en-us/library/system.icomparable(v=vs.110).aspx) interface implemented by each element of the array and by the specified value."

**Recursive Binary Search and Iterative Binary Search**

In each step, it compares the search key with the value of the middle element of the array. The keys matching in step 1 means, a matching element has been found and its index (or position) is returned. Else step 3 or 4. If the search key is less than the middle element, then the algorithm repeats its action on the sub-array to the left of the middle element or, If the search key is greater than the middle element, then the algorithm repeats its action on the sub-array to the right of the middle element. If the search key is not matching any of the subsequent left or right array, then it means that the key is not present in the array and a special "Nil" indication can be returned. The difference between the 2 is that one does it recursively and looping back though its own conditions, repeatedly splitting sub arrays, the other just adjusts the min and max values accordingly to shrink the search area until the target is either located, or deemed absent.

**Linear Search**

This search function will simply iterate though each entry in the array from the minimum to the maximum until it finds the target. This will be slow and potentially anomalous, as the result will be purely dependant on the location of the target in the array, if it is early in the array, it will be quick, if it is toward the end of the array, it will be slow.

**Test Log**

The outputted text file was saved to a temp folder on the C: drive. This text file was then converted to a Comma-separated Values (.csv) file and inserted into Excel. The results of each algorithm were then graphed for an easy visual comparison.

However, due to the significant latency of the linear search in comparison to the binary searches, illustrating all 4 examples on the same graph is difficult as seen in the above example. The 3 faster searches meld into each other and become barley distinguishable from each other. In light of this, I thought it best to separate the linear search and compare the others.

**Analysis**

The data correlated suggests that the .NET library built-in binary search function is the most efficient and contains the least anomalies. The Linear search is by far the least efficient, as to see even see the data of the binary searches effectively, it needed to be separated. The recursive search is by far the most anomalous and varying significantly in its search times. It is also the slowest of the binary search functions.

**Summary and conclusion**

The Linear search is by far the worst performer of the 4, and as such shall be excluded from any consideration. The recursive binary search is the next weakest performer, this is presumed to be caused by the amount of code that needs to be processed through each recursion. The iterative binary search was the second best performer, and not hugely slower than the build in search, this may indicate that the Array search API may include an iterative binary search. The test results do show a clear winner, the speedy Array search API.

**Appendix A**

**Test results table with time in nanoseconds.**

|  |  |  |  |
| --- | --- | --- | --- |
| Date: 08/11/2017 | Time: 11:00am | Search testing | Array size:10000 |
|  |  |  |  |
| Builtin Binary Search | Iterative binary Search | Recursive binary Search | Linear Search |
| 229.67 | 337.23 | 606.79 | 21110.57 |
| 176.27 | 250.2 | 423.54 | 21330 |
| 212.76 | 298.73 | 592.16 | 21518.01 |
| 174.26 | 249.25 | 429.41 | 19849.93 |
| 183.99 | 259.53 | 430.86 | 21142.62 |
| 174.35 | 246.9 | 415.08 | 19302.25 |
| 172.93 | 248.38 | 415.82 | 19467.84 |
| 193.06 | 248.91 | 418.79 | 19786.99 |
| 176.42 | 250.39 | 418.91 | 18900.22 |
| 175.4 | 259.9 | 422.62 | 19076.62 |
| 173.12 | 392.39 | 413.29 | 18965.82 |
| 173 | 246 | 419.71 | 19682.33 |
| 180.22 | 247.33 | 461.58 | 20200.53 |
| 172.96 | 252.06 | 420.02 | 19669.55 |
| 174.54 | 247.27 | 413.57 | 19041.58 |
| 173.03 | 244.12 | 422.46 | 19035.84 |
| 172.72 | 244.09 | 413.11 | 19096.68 |
| 172.84 | 244.58 | 415.58 | 19114.53 |
| 174.11 | 246.93 | 413.14 | 18888.89 |
| 172.78 | 246.9 | 412 | 18892.23 |
| 173.64 | 246.19 | 421.78 | 19042.2 |
| 173.83 | 244.74 | 412.27 | 18823.91 |
| 183.03 | 248.54 | 429.07 | 18993.76 |
| 176.3 | 244.92 | 413.23 | 18855.61 |
| 173.43 | 244.34 | 413.26 | 18926.65 |
| 180.9 | 246.65 | 432.77 | 19336.86 |
| 173.43 | 248.78 | 413.2 | 19562.19 |
| 181.33 | 246.07 | 421.1 | 19193.19 |
| 173.09 | 249.46 | 412.92 | 19082.48 |
| 172.87 | 245.29 | 412.03 | 19192.2 |
| 174.88 | 245.54 | 422.52 | 19736.48 |
| 178.21 | 245.17 | 427.77 | 19849.65 |
| 183.71 | 255.94 | 428.64 | 19029.66 |
| 191.83 | 319.63 | 528.94 | 19043.43 |
| 173.3 | 245.36 | 425.21 | 19232.21 |
| 173.89 | 246.1 | 413.73 | 19066.46 |
| 175.71 | 249.77 | 413.73 | 20502.94 |
| 192.51 | 256.62 | 563.14 | 21312.31 |
| 174.97 | 251.35 | 417.4 | 19080.88 |
| 172.81 | 245.82 | 414.13 | 19112.64 |
| 174.45 | 246.62 | 417.46 | 18910.93 |
| 173.92 | 245.23 | 413.79 | 19285.95 |
| 173.3 | 247.42 | 413.05 | 18856.2 |
| 173.98 | 245.26 | 412.68 | 18972 |
| 173.03 | 248.97 | 427.56 | 19502.36 |
| 174.91 | 247.76 | 417.4 | 19022.35 |
| 173.95 | 247.39 | 413.63 | 18915.56 |
| 179.48 | 248.44 | 419.56 | 19190.75 |
| 172.25 | 245.79 | 413.94 | 19491.86 |
| 172.84 | 251.44 | 422.22 | 19424.44 |
| 172.93 | 247.46 | 418.54 | 19132.99 |
| 173.15 | 253.57 | 432.22 | 19034.51 |
| 173.09 | 244.8 | 412.49 | 19013.7 |
| 173.74 | 269.68 | 425.77 | 18911.12 |
| 173.61 | 262.4 | 422.22 | 19743.27 |
| 175.8 | 246.25 | 416.2 | 18880.53 |
| 172.29 | 245.39 | 421.2 | 19200.6 |
| 177.29 | 245.97 | 420.67 | 18883.03 |
| 172.53 | 245.29 | 429.04 | 19361.4 |
| 176.14 | 260.98 | 440.49 | 19283.67 |
| 185.34 | 246.1 | 431.35 | 19093.32 |
| 173.15 | 253.38 | 418.54 | 19150.34 |
| 175.4 | 245.29 | 418.08 | 20039.91 |
| 179.48 | 303.36 | 573.54 | 20305.43 |
| 172.35 | 245.05 | 424.5 | 18884.54 |
| 172.59 | 250.02 | 426.51 | 18968.14 |
| 173.95 | 247.09 | 424.38 | 19267.77 |
| 171.98 | 244.83 | 412.83 | 18710.15 |
| 176.14 | 245.76 | 412.83 | 18991.72 |
| 174.66 | 245.33 | 413.51 | 19168.09 |
| 183.28 | 246.5 | 419.75 | 19162.65 |
| 173.37 | 245.26 | 430.27 | 19307.93 |
| 172.84 | 243.78 | 413.17 | 19179.2 |
| 173.37 | 244.21 | 413.54 | 19599.32 |
| 175.87 | 249.06 | 418.51 | 18985.36 |
| 172.69 | 245.85 | 414.93 | 18628.56 |
| 172.56 | 245.6 | 414.9 | 19001.48 |
| 171.82 | 245.91 | 412.8 | 18917.36 |
| 174.94 | 245.73 | 462.66 | 19327.69 |
| 176.36 | 261.35 | 424.68 | 19527.73 |
| 172.62 | 245.02 | 412.71 | 18773.93 |
| 173.33 | 246.04 | 415.02 | 18913.59 |
| 174.04 | 245.57 | 428.57 | 18924.67 |
| 181.21 | 249.59 | 432.99 | 19215.44 |
| 172.13 | 277.52 | 421.2 | 19575.98 |
| 222.11 | 290.89 | 417.21 | 19516.77 |
| 181.05 | 251.56 | 411.72 | 19291.88 |
| 238.35 | 317.59 | 415.73 | 20840.76 |
| 188.77 | 248.23 | 422.52 | 21000.36 |
| 201.46 | 270.45 | 505.81 | 19254.19 |
| 174.14 | 261.47 | 437.19 | 19230.23 |
| 175.28 | 248.78 | 414.44 | 19013.09 |
| 175.46 | 253.48 | 416.69 | 18844.5 |
| 182.69 | 248.29 | 428.45 | 19545.82 |
| 196.55 | 282.62 | 454.44 | 19023.4 |
| 186.52 | 286.66 | 508.84 | 19468.52 |
| 225.78 | 304.35 | 585.71 | 19123.88 |
| 180.37 | 245.97 | 413.94 | 19436.39 |
| 184.05 | 305.46 | 583.21 | 21175.06 |
| 175.53 | 269.59 | 447.13 | 18964.37 |