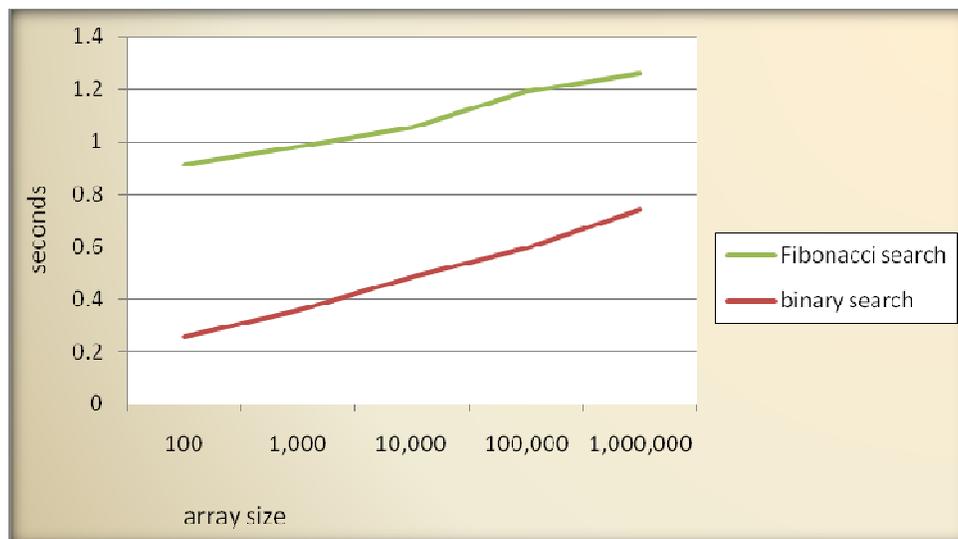


New Search Routines

The three new search routines included at Mark 22 of the NAG Fortran Library and the NAG Toolbox for MATLAB and Mark 9 of the NAG C Library are not standard search tools which return true if the sought-after item is found or false otherwise. They were created with statistical problems in mind. They examine a vector and return the index of the first value equal to the sought-after item. If this value cannot be found, the index of the immediate lower value is returned and if there is no lower value, 0 is returned. In statistics this kind of searching utility proves useful when categorising data into particular groups, for example when creating a frequency table from raw data or generating series of random numbers from a general discrete distribution. The new search routines (M01NAF, M01NBF and M01NCF) search a vector of double precision numbers, integers and character data, respectively, so there is a wide range of data they can be used on.



The results for 200 pseudorandom numbers, each sought for 10,000 times

Different algorithms were compared to find the quickest searching method. As can be seen from the graph above, relative to a binary search, the Fibonacci search appears to be surprisingly slow for quite big data sets. However, based on its more gradual increase we can assume that it can be more efficient than the binary search for the much bigger vectors.

For current computational needs, the new search functions were based on Professor Niklaus Wirth's implementation¹ of the Binary Search algorithm:

```
LEFT := 0; RIGHT := N;  
WHILE LEFT < RIGHT DO  
  mid := (LEFT+RIGHT) DIV 2;  
  IF a[mid] < x THEN LEFT := mid+1 ELSE RIGHT := mid END  
END
```

This version takes more iterations than variants that terminate when the item is found, but each time the vector is accessed only once, making this method generally more efficient.

¹ Wirth N (2004) Algorithms and Data Structures 35–36 Prentice Hall