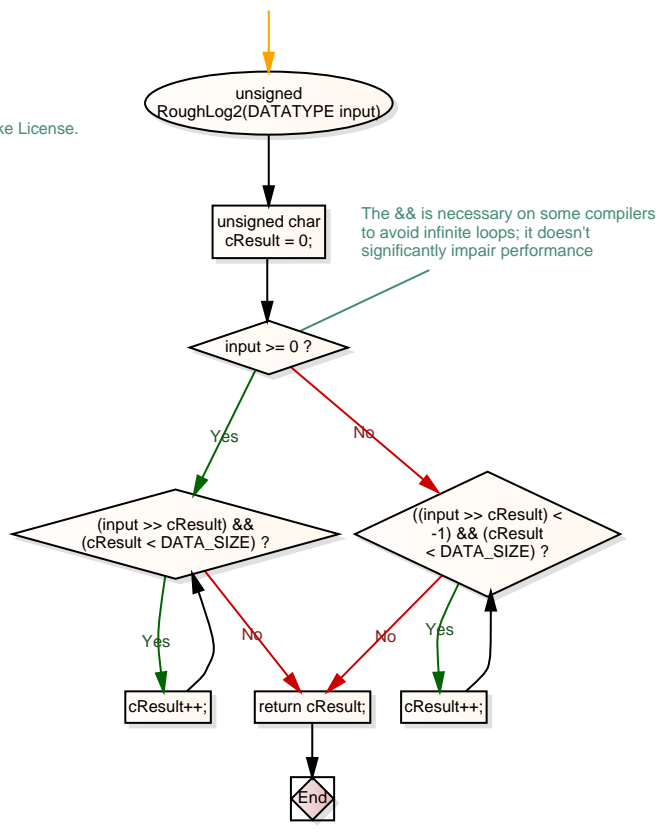
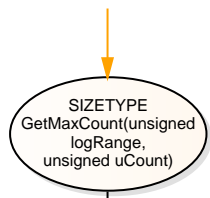




Spreadsor in C
Source: <http://en.wikipedia.org/wiki/Spreadsor>
This code is licensed under the Creative Commons Attribution-ShareAlike License.
It is from the Wikipedia article "Spreadsor" dated 2009-06-01.
Spreadsor is a sorting algorithm invented by Steven J. Ross in 2002.
It combines concepts from distribution-based sorts, such as radix sort and bucket sort, with partitioning concepts from comparison sorts such as quicksort and mergesort. In experimental results it was shown to be highly efficient, often outperforming traditional algorithms such as quicksort, particularly on distributions exhibiting structure.

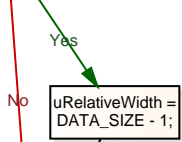
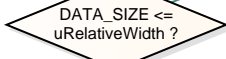




```

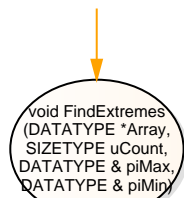
    unsigned logSize = RoughLog2Size(uCount);
    unsigned uRelativeWidth = (LOG_CONST * logRange)/((logSize
    > MAX_SPLITS) ? MAX_SPLITS : logSize);
  
```

Don't try to bitshift more than the size of an element



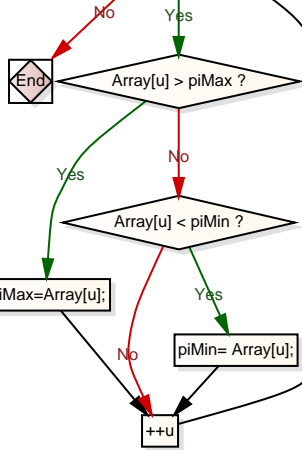
```

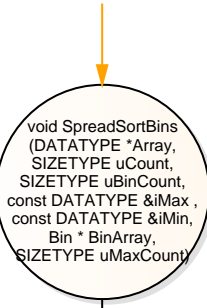
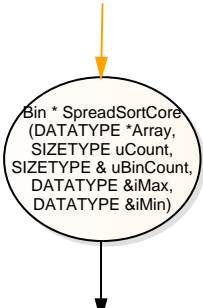
    return 1 << ((uRelativeWidth < (LOG_MEAN_BIN_SIZE + LOG_MIN_SPLIT_COUNT)) ? (LOG_MEAN_BIN_SIZE + LOG_MIN_SPLIT_COUNT) : uRelativeWidth);
  
```



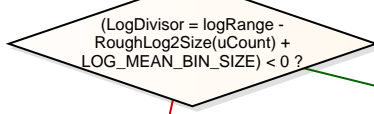
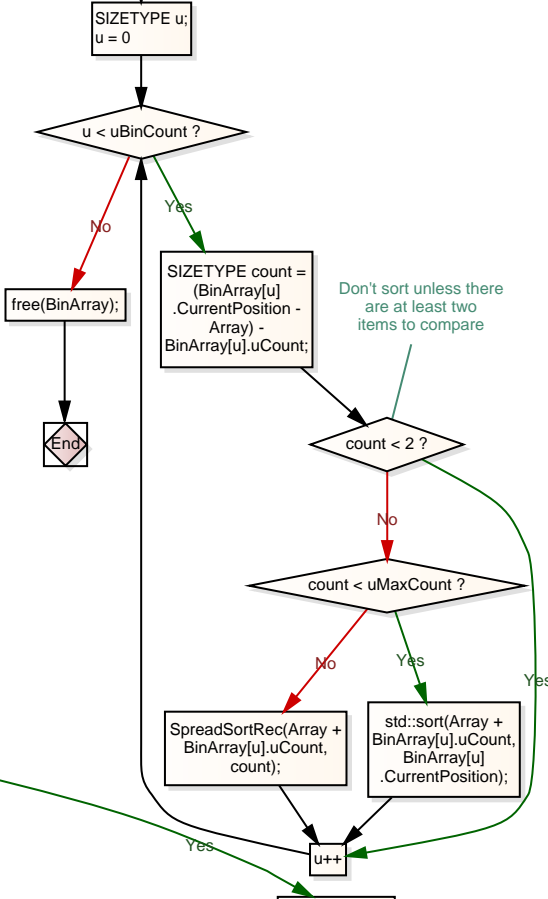
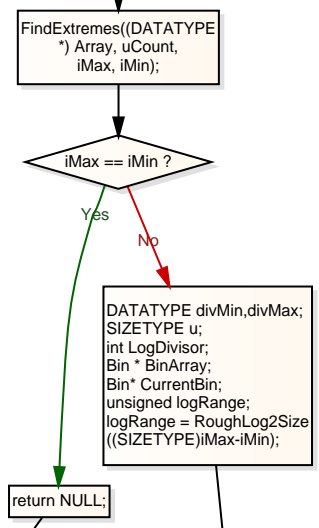
```

    SIZE_TTYPE u = 1;
    piMin = piMax = Array[0];
  
```

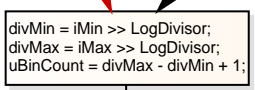
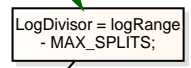
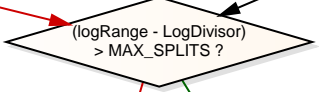




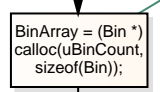
-----SpreadSort Source-----
 This step is roughly 10% of runtime but it helps avoid worst-case behavior and improves behavior with real data. If you know the maximum and minimum ahead of time, you can pass those values in and skip this step for the first iteration



The below if statement is only necessary on systems with high memory latency relative to processor speed (most modern processors)



Allocate the bins and determine their sizes



Memory allocation failure check and clean return with sorted results



Calculating the size of each bin; this takes roughly 10% of runtime

