

## Cube Root of 2 Sort

I found this sort algorithm years ago on Saxon BBS as a Public Domain contribution (remember that?).

### sort

It has proved to be the fastest sort when programmed in Basic in similar fashion - i.e. not using machine code sorts which will inherently be fast.

There are several methods of sorting, two of the most popular are the Bubble sort and the Shell sort..

Bubble sorts always work but they have to go through the entire range over and over again which makes them very slow. Shell sorts are much faster but they have a critical failure point in that if the data is arranged in just the right way, the sort can actually fail.

Below is a routine which is based on the CR2 article further below. I have used it time and again and it is Fast - even on huge data sets (10 million items), even to the point that it'll have you checking because you won't believe it did it.

to follow

the original text file

## CR2SORT.DOC

```
'
'CR2SORT DOCUMENTATION-- -6/19/1988
'
'
' THIS SORT IS A RESULT OF AN INTEREST I DEVELOPED IN SORT PROGRAMS
'AND IS MY ATTEMPT AT WRITING AN EFFICIENT ALGORITHM.
'
' THE VALUE USED TO DETERMINE COMPARE/SWAP INTERVALS IS THE CUBE
'ROOT OF 2, HENCE THE NAME 'CR2SORT'.
'
'THE SORT FUNCTIONS IN THE FOLLOWING MANNER
'
'1. THE LIST TO BE SORTED IS READ INTO AND ARRAY WITH (T) BEING EQUAL TO
'THE TOTAL NUMBER OF ELEMENTS IN THE ARRAY.
'
' 2. THE TOTAL (T) IS DIVIDED BY 2 (INTEGER DIVISION) THEN REPEATEDLY
'DIVIDED BY (P#) THE 'CR2' UNTIL THE RESULTANT (Y#) IS LESS THAN 2.
'THIS IS ACCOMPLISHED BY A WHILE/WEND LOOP. THIS IS THE SEED NUMBER
'WHICH IS USED TO BALANCE THE SORT.
'
' 3. THIS STARTING NUMBER (Y#) IS MULTIPLIED BY (P#) THE 'CR2' AND DIVIDED
'INTO THE TOTAL NO. OF ELEMENTS (T), TO OBTAIN THE VALUE (X) FOR THE
'COMPARE/SWAP INTERVAL. THIS VALUE IS OBTAINED BY INTEGER DIVISION.
```

```
'  
' 4. THE INTEGER X IS THEN USED IN A FOR-NEXT LOOP FOR COMPARING B+I  
'AND X+I ELEMENTS UNTIL THE (I) COUNT REACHES T-X. THE VALUE FOR X  
'DECREASES AS STEP 3 IS REPEATED UNTIL X REACHES 2 AND ADJACENT  
'ELEMENTS ARE BEING COMPARED. THIS IS ACCOMPLISHED WITH A WHILE/WEND  
'LOOP.  
'  
' BECAUSE THE STARTING NUMBER IS DETERMINED BY THE LENGTH OF THE  
'ARRAY, A PERFECT SORT IS OBTAINED WITHOUT SETTING A FLAG TO CHECK IF  
'A SWAP HAS BEEN MADE AND MAKING ADDITIONAL LOOPS TO COMPLETE THE SORT.  
'  
' I HAVE FOUND THIS ALGORITHM TO BE VERY EFFICIENT AND TO HAVE  
'EXCELLENT SPEED WITH A MINIMAL NUMBER OF SWAPS PERFORMED. THIS SORT  
'IS 17 OR MORE TIMES FASTER THAN A COMPARABLE BUBBLE SORT AND IS ALSO  
'FASTER THAN  
'THE SHELL SORT THAT I TESTED IT AGAINST. IT MADE FEWER COMPARES AND  
'HAD BETTER RUN TIMES. ON A TEST LIST OF 5322 STRING ELEMENTS IT MADE  
'52 PERCENT AS MANY SWAPS AND RAN IN 29 PERCENT OF THE TIME THAT IT  
'TOOK TO SORT THE LIST WITH A VERSION OF THE SHELL SORT THAT I HAVE.  
'  
' I CHECKED THIS SORT ON MANY LISTS AND FOUND NO ERRORS. I USED TEST  
'LISTS OF COMPUTER GENERATED NONSENSE WORDS FROM 2 TO 8 CHARACTERS LONG  
'WITH SUCCESSIVE NUMBERS FOR THE INTEGER ARRAY.  
'  
' IT IS A VERY SHORT ROUTINE AND SIMPLE TO WRITE INTO YOUR OWN  
'PROGRAMS OR TO USE AS A SUBROUTINE.  
'  
' THIS PROGRAM MAY BE COPIED FOR PERSONAL USE ONLY AND MAY NOT BE  
'USED AS A PART OF OTHER MARKETABLE PROGRAMS WITHOUT PERMISSION.  
'  
' IF UPLOADED TO ANOTHER BULLETIN BOARD OR PASSED ALONG TO SOMEONE  
'ELSE, PLEASE KEEP CR2SORT.DOC TOGETHER WITH THE CR2SORT PROGRAM IN  
'THE CR2SORT.ARC VERSION. (couldn't support defunct archive format - sorry)  
'  
'DAVID J. HANSON  
'1213 SOUTH 21ST STREET  
'FARGO, NORTH DAKOTA 58103  
'
```

## CR2SORT.BAS

```
'  
'SUB CR2SORT (W$(1),N%(1),T%) STATIC  
'P#=1.2599211: X=2: B=1: Y#=T\2: WHILE Y#>2: Y#=Y#/P#: WEND  
'WHILE X>1: Y#=Y#*P#: X=T\Y#  
'  FOR I=0 TO T-X  
'    IF W$(B+I)>W$(X+I) THEN SWAP W$(B+I),W$(X+I):SWAP N%(B+I),N%(X+I)  
'  NEXT I  
'WEND
```

'END SUB

From:

<http://fruitoftheshed.com/wiki/> - **FotS**

Permanent link:

[http://fruitoftheshed.com/wiki/doku.php?id=circuit\\_ideas:cube\\_root\\_of\\_2\\_sort](http://fruitoftheshed.com/wiki/doku.php?id=circuit_ideas:cube_root_of_2_sort)

Last update: **2024/02/24 17:40**

