Bucket sort (C)

From LiteratePrograms

> **Other implementations**: **C** | Java | PHP | Python

Bucket sort is possibly the simplest distribution sorting algorithm. The essential requirement is that the size of the array from which the elements to be sorted are drawn is a small, fixed constant, say n.

```
<<sort function>>=
void bucketSort(int array[], int n) {
  int i, j;
  int count[n];
  initialize counters
  count amount of each array-number
  rearrange order of array
}
```

For example, suppose that we are sorting elements drawn from {0, 1, . . ., n-1}. The integers in this set can be of the range 0 until (n-1). Bucket sort uses n counters. These counters first have to be initialized:

```
<<initialize counters>>=
for(i=0; i < n; i++) {
  count[i] = 0;
}
```

To we can determine the amount of each number in the array we want to sort.

```
<<count amount of each array-number>>=
for(i=0; i < n; i++) {
  (count[array[i]])++;
}
```

After we've determined all needed information about the array we want to sort. The array can be sorted. `j` holds the current position in the array to sort. `i` holds the current position in the counter array,

```
<<rearrange order of array>>=
for(i=0,j=0; i < n; i++) {
  for(; count[i]>0; (count[i])--) {
    array[j++] = i;
  }
}
```

Putting it together

To check the bucket sort, we output the data:

```
<<output the array>>=
for (i = 0;i < n;i++) {
  printf("%d ", array[i]);
```

```
}
printf("\n");
```

Putting it all together with a example array.

```
<<bucketsort.c>>=
#include <stdio.h>

sort function

int main() {
  int array[] = {1,3,4,6,4,2,9,1,2,9};
  int n = 10;
  int i;

  output the array

  bucketSort(array, n);

  output the array

  return 0;
}
```

This generates following output:

```
1 3 4 6 4 2 9 1 2 9
1 1 2 2 3 4 4 6 9 9
```

Download code (http://en.literateprograms.org/index.php?
title=Special:Downloadcode/Bucket_sort_(C)&oldid=11824)

Retrieved from "http://en.literateprograms.org/index.php?title=Bucket_sort_(C)&oldid=11824"
Categories: Bucket sort │ Programming language:C │ Environment:Portable

- This page was last modified on 27 December 2007, at 18:03.
- Content is available under the MIT/X11 License.