

Re: array random sort

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2005-11/msg01520.html>

- *From:* "James Curran" <jamescurran@xxxxxxxx>
 - *Date:* Thu, 17 Nov 2005 19:04:38 -0500
-

The problem with Fred's solution is that while it appears to be an $O(n)$ algorithm, `RemoveAt` itself is $O(n)$, making the overall algorithm $O(n^2)$. If the input array get big, it's going to get very slow.

The trick is to avoid deleting anything. We're starting with an array of N , and we finish with an array of N , so there's no reason to delete anything. This version is really $O(n)$:

```
// Shuffles inplace.
public static List<T> shuffleList(List<T> listToShuffle)
{
    for (int k = listToShuffle.Count-1; k > 1; --k)
    {
        int randIndx = Common.rand.Next(k); //
        T temp = listToShuffle[k];
        listToShuffle[k] = listToShuffle[randIndx]; // move random num to
        end of list.
        listToShuffle[randIndx] = temp;
    }
    return listToShuffle;
}
```

So, say, for example, `listToShuffle` has 10 elements (0-9). First we pick a random number between 0 & 8, and we swap the element at the position with the element at position 9. Then we pick an number between 0 & 7, and swap that element with [8]. And so on until we're pick a number between 0 & 1 and swapping it with `listToShuffle[2]`

—
Truth,
James Curran
[erstwhile VC++ MVP]

Home: www.noveltheory.com Work: www.njtheater.com
Blog: www.honestillusion.com Day Job: www.partsearch.com

"Fred Mellender" <nospamPlease_fredm@xxxxxxxxxxxxxxxxxxxx> wrote in message
[news:3h4ff.13497\\$cg.7381@xxxxxxxxxxxxxxxxxxxx](mailto:news:3h4ff.13497$cg.7381@xxxxxxxxxxxxxxxxxxxx)
>

Re: array random sort

```
> "gl" <gl@xxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message
> <news:7E0D71F9-962F-4A97-9770-EC71E6659A33@xxxxxxxxxxxxxxxxxxxx>
>> How do I take an array or arraylist, and sort it randomly? Like suppose
>> the
>> items in it are (1,2,3,4,5) and I want to get it to be in a random order
>> (2,3,1,4,5). How do you do that? I think it's a call to the array or
array
>> lists sort method, but i'm not exactly sure how you do it.
>
>
> public static List<T> shuffledList(List<T> listToShuffle)
> {
> /*
> * Make a new list of elements picked from listToShuffle
> * in a random order.
> */
>
> List<int> ints = new List<int>(listToShuffle.Count); //0, 1,
> 2, ...
> for (int i = 0; i < listToShuffle.Count; i++)
> ints.Add(i);
>
> List<T> randList = new List<T>(listToShuffle.Capacity);
>
> for (int k = 0; k < listToShuffle.Count; k++)
> {
> int randIndx = Common.rand.Next(ints.Count); //random
> from 0, 2, .. not already picked
> int randK = ints[randIndx];
> randList.Add(listToShuffle[randK]);
> ints.RemoveAt(randIndx);
> }
>
> return randList;
> }
>
>
```

• **References:**

◆ **Re: array random sort**

◇ From: Fred Mellender

- Prev by Date: **Replace**
- Next by Date: **Re: C# 2005 not packaged without Visual Studio?**
- Previous by thread: **Re: array random sort**
- Next by thread: **Re: array random sort**

Re: array random sort

- Index(es):
 - ◆ *Date*
 - ◆ *Thread*