Re: Help me with references

Source: http://coding.derkeiler.com/Archive/Java/comp.lang.java.help/2005−08/msg00413.html

- *From*: "Oliver Wong" <owong@xxxxxxxxxxxxx>
- *Date*: Tue, 16 Aug 2005 18:33:24 GMT

"Kenneth P. Turvey" <kt@xxxxxxxxxxxxxxxxx> wrote in message
news:f6a9t2−7b6.ln1@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
> Oliver Wong wrote:
>> If you're interested though, I can still answer the questions in your
>> post.
>
> I am interested. I can't think of any situation in which bubble sort is
> the
> best choice for a sorting algorithm in a real machine.

That's not the questions I was referring to, but I'll add it to the list
of questions I'll try to address.

> If you know you are going to have fewer than 10 elements
> why are you writing your own sorting routine?

Knowing that you have fewer than 10 elements is a particularly good
situation to write your own sorting algorithm, because you have knowledge
about the problemspace that the general quicksort algorithm doesn't.
Quicksort is optimized for large Ns. In the case where you not only know
that your N is not large, but you additionally know the exact value of N,
you can design an algorithm that is optimal for that specific N.

> Then there is also the question of why you picked bubble sort from other
> sorts that are O(n^2), like insertion sort, which is simpler and faster.

I picked Bubblesort because your original post implied something along
the lines of "it never makes sense to use bubblesort". I agree that a hybrid
quicksort/insertion sort would be even better than a hybrid
quicksort/bubblesort. I just wanted to point out that a hybrid
quicksort/bubblesort is better than a pure quicksort.

> Right, but if you know your sort will be done in no time why are you
> spending time optimizing it.

"No time" is dependent on the requirements of the environment. If you're
workign with real time systems and you know for a fact that if the sort
takes more than 10 milliseconds, for example, sick patients in hospital will
not receive the proper injection of medication in time and die. Or aircrafts

will collide with each other, possibily killing hundreds. 11 milliseconds, which may seem like "no time" in most applications, isn't good enough. Optimizing it down to 8 milliseconds is worth it.

> I am interested. I can't think of any situation in which bubble sort is
> the
> best choice for a sorting algorithm in a real machine.

Depending on your definition of "best" (smallest worst case asymptotic running time?) and "real" (made by a big company such as Intel, AMD, IBM or Sun?), I'll probably agree with you. I could try to construct a situation where the desired output is not actually a sorted array, but an "almost sorted array" with some sort of rigorous definition of "almost sorted" such that BubbleSort might achieve this in (n^2+n)/4 comparisons whereas InsertionSort can only do it after the full (n^2+n)/2 in the worst case, but quite frankly I'm too lazy.

Note that pragmatically, "best" may not always be synonymous with "technically best", in the sense that factors other than technology (e.g. business, politics, etc.) may affect what algorithms are used in a given environment.

I'm working with a client which wants a file management application that handles proper synchronization and locking of files with multiple concurrent users. Except the client wants the program written in VBScript (which is only slightly more powerful than batch files, has no file locking support, no synchronization support, and no threading support). Obviousy, VBScript is not the best tool for the job, from a technical point of view, but the client is writing the cheque and that's what they want. So that's what we do.

The point of this anecdote is that you and I, as rational software developers, are not able to see an application of the BubbleSort algorithm where it is the optimal tool for a problem − but that does not nescessarily imply that there will *NEVER* (emphasizing this keyword) be a situation where it might make sense to use Bubblesort.

− Oliver

.

- ***Follow−Ups***:
  - ♦ ***Re: Help me with references***
    - ◊ *From:* Kenneth P. Turvey

- ***References***:
  - ♦ ***Help me with references***
    - ◊ *From:* Masamunexiii

- ***Re: Help me with references***
  - ◊ *From:* Kenneth P. Turvey
- ***Re: Help me with references***
  - ◊ *From:* Oliver Wong
- ***Re: Help me with references***
  - ◊ *From:* Kenneth P. Turvey
- ***Re: Help me with references***
  - ◊ *From:* Oliver Wong
- ***Re: Help me with references***
  - ◊ *From:* Kenneth P. Turvey

- Prev by Date: ***Re: Using ImageObserver instead of MediaTracker***
- Next by Date: ***Re: Help me with references***
- Previous by thread: ***Re: Help me with references***
- Next by thread: ***Re: Help me with references***
- Index(es):
  - ***Date***
  - ***Thread***