Subject: Stable Sort algorithm
Posted by forlano on Thu, 20 Feb 2025 16:01:56 GMT
View Forum Message <> Reply to Message

Hello,

do we have in core a STABLE fast sort algorithm?

Thanks,
Luigi

Subject: Re: Stable Sort algorithm
Posted by koldo on Thu, 20 Feb 2025 16:24:52 GMT
View Forum Message <> Reply to Message

In https://www.ultimatepp.org/srcdoc$Core$Tutorial$en-us.html it appears this:

File Attachments
1) firefox_0JTROk3tWL.png, downloaded 213 times

Subject: Re: Stable Sort algorithm
Posted by JeyCi on Thu, 20 Mar 2025 06:28:49 GMT
View Forum Message <> Reply to Message

can just advise to use COM-dll from bedvit (if it will connect to Upp)
1. ArraySort (String)
2. ArraySort (Variant)

Subject: Re: Stable Sort algorithm
Posted by Didier on Thu, 20 Mar 2025 18:48:03 GMT
View Forum Message <> Reply to Message

Recently I had challenged sort algorithms: Upp, Std, other ...And Upp sorts were the fastest :) (at least for my use-case)

If you want to transform an unstable alogrithm into stable one, you just need to add data index comparison in you're comparison criteria.
I tried it with Upp sort, and it was faster than stable sort :?: (for my use case were there where the comparison criteria was quite complex)

Subject: Re: Stable Sort algorithm
Posted by JeyCi on Mon, 24 Mar 2025 21:05:36 GMT
View Forum Message <> Reply to Message

Didier wrote on Thu, 20 March 2025 19:48
If you want to transform an unstable alogrithm into stable one, you just need to add data index comparison in you're comparison criteria.

as I remember, he took comparator from here. He is developing COM and uses sorting of pointers to memory - that is considered to be faster, but as Iknow, UPP IDE also works with stack and heap directly in its source-files, just hiding pointers to Core and other packages.

Subject: Re: Stable Sort algorithm
Posted by mirek on Fri, 18 Apr 2025 12:19:44 GMT
View Forum Message <> Reply to Message

Didier wrote on Thu, 20 March 2025 19:48Recently I had challenged sort algorithms: Upp, Std, other ...And Upp sorts were the fastest :) (at least for my use-case)

It is possible they are not faster in general case - they are tuned to sort Upp::String - which usually is the use-case.

Subject: Re: Stable Sort algorithm
Posted by Didier on Sun, 20 Apr 2025 17:50:17 GMT
View Forum Message <> Reply to Message

Maybe my use case is close to that:
Sort an array of int based on sort criterias comming from data pointed by the index: sounds pretty close to sorting strings

If m'y memory is good, the upp::sort was 2 times faster than std::sort