

Szövegrendezés szám szerint

Gyakori feladat, hogy számokat tartalmazó szövegeket kell rendeznünk természetes módon, amely azt eredményezi, hogy a szöveggént tárolt számok rendezése nem lesz megfelel: az `["1","3","21"]` számok szöveggént való rendezése után `["1","21","3"]` sorrendbe kerülnek. A szebb Java nem ad erre eszközt és sajnos tágabb körben is probléma marad ez a probléma: nincs erre megoldás még a népszerű [Jakarta Commons](#) gyjteményben sem. Ezt a problémát járta körül az [In My Opinion blog](#) írója is, s az alábbi megoldásokat találta:

- [Natural String Order](#) Stephen Friedrich blogján, ahol az implementált algoritmus teljes körű megoldást ad különféle rendezési szempontokra és nemzetközi karakterekre is.
- [HumaneStringComparator](#), Tim Fennell implementációja.
- Pierre-Luc Paur [programja](#).
- [Natural Order String Comparison](#), amely C nyelv implementáció.
- [Natural Order Numerical Sorting](#), amely cikk b lére eresztve foglalkozik a problémával, sok linket tartalmaz egyéb projektekre, de a linkek jó része vagy halott vagy nem Java megoldásra mutat.

Az oldalon készítettek egy tesztkörnyezetet, amelyben kipróbálták a felsorolt implementációkat, s az alábbi eredményeket kapták:

Pierre-Luc Paur	NaturalOrderComparator	453ms	Eléggé gyors
Stephen Friedrich	NaturalComparator	4828ms	Locale szerint rendez, de lassú
Stephen Friedrich	NaturalComparatorAscii	360ms	A leggyorsabb, de csak ASCII
Stephen Friedrich	NaturalComparatorIgnoreCaseAscii	500ms	Eléggé gyors, de case-insensitive, de csak ASCII
Tim Fennell	HumaneStringComparator	4797ms	Oktatásra kiváló
Natív Java	<code>String.compareTo()</code>	235ms	-