

# TÖL403G - Greining Reiknirita - Verkefni 1

Þið megið vinna í hóp með mest þremur við þetta verkefni. Þá þurfa nöfn allra að koma fram á skýrslunni sem skilað er.

## Lýsing

Í þessu verkefni á að nota forritasafnið `Sorter` sem gefið er sem þýdd java skrá. Í klasanum `Sorter` eru 10 röðunaraðferðir sem heita `s1`, `...`, `s10` og taka allar inn viðfang af taginu `int[]` a. Allar röðunaraðferðirnar raða jákvæðum heiltölum í vaxandi röð.

Röðunaraðferðirnar sem eru útfærðar eru nákvæmlega eins og lýst er í kennslubókinni og eru

- Insertion sort
- Merge sort
- Heap sort
- Quick sort með vendistak valið sem
  - aftasta stak,  $n$
  - miðju stak,  $n/2$
  - miðgildið af fremsta, miðju og aftasta staki
  - slembistak
- Counting sort
- Radix sort
- Bucket sort

Quicksort aðferðunum hefur verið breytt þannig að ekki fæst  $\Omega(n^2)$  keyrslutími ef öll stökin eru eins, ef öll stökin eru ólík fæst nákvæmlega sama keyrsla og í bók. Partition fallið sem notað er er gefið á síðunni.

Sumar aðferðir gætu tekið óeðlilega langan tíma eða notað of mikið minni.

Verkefnið gengur út á að finna hvaða aðferð er hvað án þess að kíkja á þýdda bætakóðann. Skrifid skýrslu þar sem rökstutt er hvert val með því að vísa til

- Tímamælinga fyrir besta, versta, meðaltals tilfelli.
- Minnisnotkunar, þ.e. hve mikið minni er notað aukalega
- Hvort reikniritið er endurkvæmt eða ekki.

Til að finna tímaflækju verður að nota margar mismunandi stærðir og lítil gildi á  $n$  gefa ónákvæmar mælingar, rökstyðja verður allt mat á stærðargráðu útreikningum og ætti a.m.k. að nota 4-5 gagnapunkta til að meta vaxtarhraða.

Í java má finna út minnið með því að skoða `Runtime.getRuntime().freeMemory()` og `Runtime.getRuntime().totalMemory()`, sjá t.d. <http://stackoverflow.com/questions/3021727/obtaining-memory-available-to-jvm-at-runtime>. Til að mæla tíma er best að nota `System.nanoTime()`, sjá t.d. <http://stackoverflow.com/questions/180158/how-do-i-time-a-methods-execution-in-java>.

Athugið að `Sorter.class` skráin ætti að virka með java 1.5 og nýrra en þó er öruggara að nota java 1.6.

## Skil

Skilið forriti sem þið notið við mælingar (annað hvort java, shell script eða bæði) og skýrslu sem **pdf skjali** (allt annað verður ekki lesið). Skilið pakkanum, annað hvort sem .zip eða .tar.gz, á skilaskrá á uglunni.

Skilafrestur er til miðnættis sunnudaginn 2. febrúar.

## Skýrsla

Í skýrslu á að koma fram hvaða mælingar voru framkvæmdar og hvaða ályktanir þið dragið af þeim. Þið megið gera ráð fyrir að lesandi þekki allra röðunaraðferðirnar en þið verðið að lýsa því hvaða inntak var notað, hverjar voru niðurstöður mælinga og hvaða þýðingu það hefur miðað við fræðin. Hafið eftirfarandi í huga þegar þið gerið skýrsluna.

- Styttri texti er betri en lengri
- Forrit lengri en 10 línur eiga ekki heima í skýrslu
- Langir listar af tölum eru slæmir, töflur eru betri
- Plot eru betri en töflur
- Log-log plot eru allra best til að greina á milli margliða
- Það þarf að setja niður nægilegar upplýsingar til að lesandi geti endurtekið tilraunina án þess að hafa forritin ykkar.
- Skýrslan á ekki að vera skrifuð í þeirri röð sem þið framkvæmduð prófanir.

Einkunn verður gefin fyrir skýrsluna vegur læsileiki skýrslunnar hátt í þeirri einkunn.