

Schmekerezada - being tetra-threaded Quicksort 'Magnetica' - redeeming Schmekerism - in search for FASTNESS... - FASTEST sort console tool?!

A short review by samayce@samayce.com, 2022-Oct-18



Schmekerezada - being tetra-threaded Quicksort 'Magnetica' - redeeming Schmekerism - in search for FASTNESS... - FASTEST sort console tool?!

A short review by samayce@samayce.com, 2022-Oct-18



Schmekerezada - being tetra-threaded Quicksort 'Magnetica' - redeeming Schmekerism - in search for FASTNESS... - FASTEST sort console tool?!

A short review by samayce@samayce.com, 2022-Oct-18



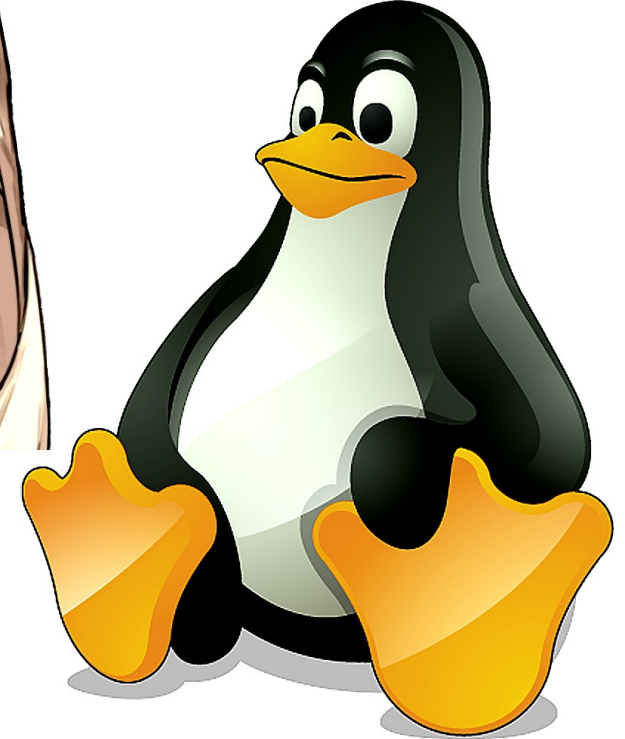
Schmekeriada - being tetra-threaded Quicksort 'Magnetica' - redeeming Schmekerism - in search for FASTNESS... - FASTEST sort console tool?!

A short review by sanmayce@sanmayce.com, 2022-Oct-12



- 100% FREE (free as free beer), open-source, written in C;
- Linux and Windows binaries - GCC, CLANG and ICL compilable;
- Sorting Japanese Wikipedia XML dump (15,674,562,752 bytes) at 1,262,348 Lines-Per-Second, on USB drive;
- Faster than Linux' sort (GNU coreutils) 8.32 tool, usually by 12%..90%;
- Sources serving as a nifty playground allowing further tweaks, EZ;
- Sorts 8[+] bytes long fixed-length keys at Schmekerian speeds;
- Scalar code at the moment, incoming AVX512 partitioning...;
- Written by machinely yours Sanmayce.

Linux' sort



Schmekeriada

This build (2022-Oct-10) features Quicksort-Magnetica-Strongfool, also buffered dump of sorted data.
This tool is 100% FREE and open-source, for improvements: sanmayce@sanmayce.com, enfun!

The full C source (along with the elf and console compile lines for Windows/Linux) is at my Internet drive: <https://drive.google.com/file/d/1boRj-y70ZKg88CPzqfQJtXCEXZFzAwue/view?usp=sharing>
<https://www.overclock.net/threads/the-definitive-quicksort-benchmark.1800616/#post-29029160>

<https://www.linuxquestions.org/questions/programming-9/qsort-vs-%27magnetica%27-quicksort-4175703333/page2.html#post6385269>
<https://forums.fedoraforum.org/showthread.php?328937-Fedora-36-vs-Windows-10-Quicksort-functions-written-in-C-running-much-slower-!&p=1861788#post1861788>

README_Schmekerezada.TXT:

README_Schmekerezada.TXT

Short DIZ (description) of the package 'Schmekerezada.tar.gz'.

This console tool allows sorting lines (both Windows and Linux) of a given file, FAST.

The compiles/binaries are targeting SSE4.2 CPUs with 1 or 4 threads, thus my cutest machinette Thinkpad 11e (8GB RAM and 4/4 cores/threads) sets the base.

The package contains these:

```
[sanmayce@djudjeto2 Schmekerezada]$ ls -l
```

```
-rwxr-xr-x. 1 sanmayce sanmayce      18675 Jul 13 00:05 Akkodah_v3.h
-rwxr-xr-x. 1 sanmayce sanmayce       1439 Jul 13 00:05 bench_PARAMETER.sh
-rwxr-xr-x. 1 sanmayce sanmayce       9327 Jul 13 00:05 crmsort.c
-rwxr-xr-x. 1 sanmayce sanmayce     10704 Jul 13 00:05 crmsort.h
-rwxr-xr-x. 1 sanmayce sanmayce       150 Jul 13 00:05 GENERATE_Xmillion_Knight-Tours.bat
-rwxr-xr-x. 1 sanmayce sanmayce       151 Jul 13 00:05 GENERATE_Xmillion_Knight-Tours.sh
-rwxr-xr-x. 1 sanmayce sanmayce     78263 Jul 13 00:05 Knight-Tour_FNV1A_YoshimitsuTRIADii_vs_CRC32_TRISMUS.c
-rwxr-xr-x. 1 sanmayce sanmayce     33792 Jul 13 00:05 Knight-Tour_FNV1A_YoshimitsuTRIADii_vs_CRC32_TRISMUS.elf
-rwxr-xr-x. 1 sanmayce sanmayce    139776 Jul 13 00:05 Knight-Tour_FNV1A_YoshimitsuTRIADii_vs_CRC32_TRISMUS.exe
-rwxr-xr-x. 1 sanmayce sanmayce     78553 Jul 13 00:05 log_su_Intel_Celeron_N4100_Cores-4.txt
-rwxr-xr-x. 1 sanmayce sanmayce     66313 Jul 13 00:05 log_su_Intel_Kaby-Lake_i5-7200U_Cores-2.txt
-rwxr-xr-x. 1 sanmayce sanmayce    151607 Jul 13 00:05 Magnetica_v18.h
-rwxr-xr-x. 1 sanmayce sanmayce       303 Jul 13 00:05 MAKE_CLANG_Schmekeriada.bat
-rwxr-xr-x. 1 sanmayce sanmayce       404 Jul 13 00:05 make_elf_CLANG_Schmekeriada.sh
-rwxr-xr-x. 1 sanmayce sanmayce       1066 Jul 13 00:05 make_elf_exe_GCC_Schmekeriada.sh
-rwxr-xr-x. 1 sanmayce sanmayce       728 Jul 13 00:05 MAKE_ICL.bat
-rwxr-xr-x. 1 sanmayce sanmayce     23515 Jul 13 00:05 quadsort.c
-rwxr-xr-x. 1 sanmayce sanmayce     12100 Jul 13 00:05 quadsort.h
-rwxr-xr-x. 1 sanmayce sanmayce    3288935 Jul 13 00:05 Quicksort_Magnetica_COVERS.pdf
-rwxr-xr-x. 1 sanmayce sanmayce 268684544 Jul 13 00:05 Schmekerezada_CLANG_16.0.1_SSE4.2_TetraThread.elf
-rwxr-xr-x. 1 sanmayce sanmayce 1606950 Jul 13 00:05 Schmekerezada_CLANG_16.0.1_SSE4.2_TetraThread.elf.asm
-rwxr-xr-x. 1 sanmayce sanmayce 268648576 Jul 13 00:05 Schmekerezada_GCC_13.0.1_SSE4.2_MonoThread.elf
-rwxr-xr-x. 1 sanmayce sanmayce 269694245 Jul 13 00:05 Schmekerezada_GCC_13.0.1_SSE4.2_MonoThread.exe
-rwxr-xr-x. 1 sanmayce sanmayce 268656936 Jul 13 00:05 Schmekerezada_GCC_13.0.1_SSE4.2_TetraThread.elf
-rwxr-xr-x. 1 sanmayce sanmayce   953727 Jul 13 00:05 Schmekerezada_GCC_13.0.1_SSE4.2_TetraThread.elf.asm
-rwxr-xr-x. 1 sanmayce sanmayce 269703091 Jul 13 00:05 Schmekerezada_GCC_13.0.1_SSE4.2_TetraThread.exe
-rwxr-xr-x. 1 sanmayce sanmayce   306350 Jul 13 00:05 Schmekeriada.c
-rwxr-xr-x. 1 sanmayce sanmayce   12927 Jul 13 00:05 sort_vs_Schmekerezada.sh
-rwxr-xr-x. 1 sanmayce sanmayce    6144 Jul 13 00:05 timer64.exe

-rwxr-xr-x. 1 sanmayce sanmayce 1359441920 Jul 13 00:05 linux-6.1.38.tar
-rwxr-xr-x. 1 sanmayce sanmayce 3313061631 Jul 13 00:05 www.ncbi.nlm.nih.gov_genome_guide_human_GRCh38_latest_genomic.fna
-rwxr-xr-x. 1 sanmayce sanmayce 3870000000 Jul 13 00:05 30000000.KnightTours.txt
-rwxr-xr-x. 1 sanmayce sanmayce 2099451904 Jul 13 00:05 Fedora-Workstation-Live-x86_64-38-1.6.iso
```

```
[sanmayce@djudjeto2 Schmekerezada]$ sha1sum *
```

```
a21deb40fec1591e8d9505c15d54322faa3e4618 Akkodah_v3.h
5af613a91318f5d15cea962123f0af44ad5c7338 bench_PARAMETER.sh
06b1f4034569ba908c7236e54bb7689d4e071121 crumsort.c
22b68f5ecc9e355e6ed9be4044d1741f255bb40b crumsort.h
413c26e8443ddadd0605bfb8d68eea795b295b20 GENERATE_Xmillion_Knight-Tours.bat
595dcc597f87bf6f71613354c000b5dfc620e07a GENERATE_Xmillion_Knight-Tours.sh
3a0be7e1767007cc2e9c0de949d767af64b48bc6 Knight-Tour_FNV1A_YoshimitsuTRIADii_vs_CRC32_TRISMUS.c
98e4d1ecda7d862604ebc2ece300d5f57ba5365b Knight-Tour_FNV1A_YoshimitsuTRIADii_vs_CRC32_TRISMUS.elf
1f5907713512e24ee75115a32e64ec8bbd9ff955 Knight-Tour_FNV1A_YoshimitsuTRIADii_vs_CRC32_TRISMUS.exe
2aa7fc07cd9981f1f62dc68799edb501124c19fb log_su_Intel_Celeron_N4100_Cores-4.txt
3a2b4e105d52cded87a163ca393a58dda969683 log_su_Intel_Kaby-Lake_i5-7200U_Cores-2.txt
dc1f49ecc858da545d4bc7c86d2d4beb683adc3d Magnetica_v18.h
42ea7db44d8821477ae250e123f51c780728954f MAKE_CLANG_Schmekeriada.bat
2319549123d6e462ab92f4fb0e9c1c01f5c61d3e make_elf_CLANG_Schmekeriada.sh
d2037c50131cae3e853968be90a2fc52d252a65c make_elf_exe_GCC_Schmekeriada.sh
b275b35d5a0d370809dc355a78e61bc12f150767 MAKE_ICL.bat
1fd6d4365743b2ddb59d167cdd2fd0f078f042de quadsort.c
3817e01a5bd6b0b929d5ce1b9ac8014de5f4c14f quadsort.h
770ee11a3d1e1176abe5d724eb3bc5ba334fab65 Quicksort_Magnetica_COVERS.pdf
8bb9012967b2c319c46e1e299e81d4583954bfe8 Schmekerezada_CLANG_16.0.1_SSE4.2_TetraThread.elf
3f593ba538a7e44f9e3c18121266b0bd6a0ce29d Schmekerezada_CLANG_16.0.1_SSE4.2_TetraThread.elf.asm
57a2789a730aa068a8977e811210418ddcd75c2c Schmekerezada_GCC_13.0.1_SSE4.2_MonoThread.elf
66b49258c127ac0f8699f0e232efaa3402792555 Schmekerezada_GCC_13.0.1_SSE4.2_MonoThread.exe
192b346bcf7a3067fe305108318d50e87629e29d Schmekerezada_GCC_13.0.1_SSE4.2_TetraThread.elf
cbc57c319311cf9b61bee238a3ae565aaafffa30 Schmekerezada_GCC_13.0.1_SSE4.2_TetraThread.elf.asm
ffcb37bf83f2c56e045c9212ce09b9dd1e01ac98 Schmekerezada_GCC_13.0.1_SSE4.2_TetraThread.exe
bc8cc0fc85a57ab64192fc741b7444ff6f3a50a0 Schmekeriada.c
7b76a8f00a45a873784a0c8ee673a0b0051b32ab sort_vs_Schmekerezada.sh
97c68b70851e3f1db82e73107849fda988308c48 timer64.exe

74f6610fe722658cf2940be1853e9c13d4da464a linux-6.1.38.tar
daa181d9ec4b58d17778bfc5bdc96472c76fd15 www.ncbi.nlm.nih.gov_genome_guide_human_GRCh38_latest_genomic.fna
11a6084ed17de2a37e05e10a89baa8b22d90e68b 300000000.KnightTours.txt
9904b33356a7852fdf0df49c6e63d74361cc5d5b Fedora-Workstation-Live-x86_64-38-1.6.iso
```

The benefits (compared to Windows' sort and Linux' sort) are:

- 100% FREE sourcecode, no licenses and shenanigans;
- Faster than both, see 'log_su_Intel_Celeron_N4100_Cores-4.txt';
- Neither of both are really cross-platform, no binaries (counterparts) found on the net that are transparently usable;
- Good starting point/playground for C coders.

Enfun!

2023-Jul-12,
Sanmayce

sort_vs_Schmekerezada.sh:

```
#
#
#
#
#
#
```

```
if [ ! -f "/30000000.KnightTours.txt" ]; then
sh GENERATE_Xmillion_Knight-Tours.sh 30000000
fi
sh bench_PARAMETER.sh linux-6.1.38.tar
sh bench_PARAMETER.sh www.ncbi.nlm.nih.gov_genome_guide_human_GRCh38_latest_genomic.fna
sh bench_PARAMETER.sh Fedora-Workstation-Live-x86_64-38-1.6.iso
sh bench_PARAMETER.sh 30000000.KnightTours.txt
```

```
#-rwxrwxrwx. 1 kaze kaze 1359441920 Jul 11 14:53 linux-6.1.38.tar
#-rwxrwxrwx. 1 kaze kaze 3313061631 Apr 8 2022 www.ncbi.nlm.nih.gov_genome_guide_human_GRCh38_latest_genomic.fna
#-rwxrwxrwx. 1 kaze kaze 2099451904 Jul 11 17:35 Fedora-Workstation-Live-x86_64-38-1.6.iso
#-rwxrwxrwx. 1 kaze kaze 3870000000 Jul 12 01:33 30000000.KnightTours.txt
```

#	Corpus Name	linux-6.1.38.tar	Human_Genome_DNA	30000000.KnightTours.txt	Fedora-Workstat...-38-1.6.iso
#	Corpus Size in Bytes	1,359,441,920	3,313,061,631	3,870,000,000	2,099,451,904
#	Sorter \ Corpus Size in Lines	35,585,653	40,902,071	30,000,000	8,189,810
#	sort v9.0 --parallel=4 -T ./	99,472,516,848 instructions 21,494,037,021 branches 461,065,737 branch-misses 20.5 seconds	128,285,206,805 instructions 27,548,013,135 branches 579,823,078 branch-misses 42.9 seconds	109,679,415,651 instructions 23,469,594,117 branches 156,337,384 branch-misses 29.8 seconds	27,716,611,486 instructions 5,918,985,732 branches 166,284,942 branch-misses 13.0 seconds
#	sort v9.0 --parallel=16 -T ./	99,826,000,759 instructions 21,496,860,784 branches 465,714,553 branch-misses 20.1 seconds	128,975,431,969 instructions 27,614,861,865 branches 587,152,024 branch-misses 39.7 seconds	106,611,922,490 instructions 22,799,055,675 branches 169,894,912 branch-misses 30.1 seconds	30,359,262,417 instructions 6,494,585,359 branches 176,038,949 branch-misses 13.4 seconds
#	Schmekerezada_CLANG_16.0.1 (v19)	64,682,963,173 instructions 13,770,680,866 branches 423,462,180 branch-misses 14.6 seconds	104,566,733,962 instructions 22,623,299,550 branches 571,367,377 branch-misses 30.5 seconds	135,489,760,619 instructions 31,235,409,067 branches 521,102,315 branch-misses 24.3 seconds	37,726,691,231 instructions 9,080,478,581 branches 120,282,275 branch-misses 9.8 seconds
#	Schmekerezada_GCC_13.0.1 (v19)	73,570,739,976 instructions 14,652,384,005 branches 421,083,216 branch-misses 19.5 seconds	120,331,569,174 instructions 24,465,263,549 branches 568,237,768 branch-misses 37.8 seconds	146,686,441,751 instructions 33,075,459,615 branches 523,361,025 branch-misses 32.3 seconds	42,112,144,823 instructions 9,627,303,775 branches 121,180,507 branch-misses 14.7 seconds

Note1: The benchmark is in 'performance' mode as superuser;

```
# Note2: Linux version 5.18.15-200.fc36.x86_64 (mockbuild@bkernel01.iad2.fedoraproject.org) (gcc (GCC) 12.1.1 20220507 (Red Hat 12.1.1-1), GNU ld version 2.37-27.fc36) #1 SMP
PREEMPT_DYNAMIC Sun Jul 31 21:30:34 UTC 2022
# Note3: Testmachine: Laptop i5-7200U CPU, 3.1GHz max turbo 2cores/4threads, L1d: 64 KiB (2 instances), L1i: 64 KiB (2 instances), L2: 512 KiB (2 instances), L3: 3 MiB (1 instance), 36GB
DDR4 2133MT/s, running Fedora 36;
# Note4: Schmekerezada is tetrathreaded;
# Note5: The current drive: SSD SATA KINGSTON SKC6001024G (1GB cache);
# Note6a: Partition in use:
# Note6b: Filesystem      Type  Size  Used Avail Use% Mounted on
# Note6c: /dev/sdb1      ext4 331G 226G  89G  72% /
# Note7: LC_ALL=C locale was used for sort;
# Note8: After sorting, checking the shasum for both outputs - they all matched;
# Note9: CLANG compiler is significantly better than GCC, too many times;
# NoteA: The KT_30M corpus is of fixed-line size - 128 bytes each line - all lines unique, here the LittleEndian-To-BigEndian technique pays off;
# NoteB: The time statistics (wall clock) are reported by Linux' perf;
# NoteC: It is worth mentioning that CLANG executable is executed before GCC counterpart, it means possible caching of the whole file is more likely for the latter.
#
# So, cumulatively:
# 14.6+30.5+24.3+9.8=79.2 seconds
# 20.1+39.7+30.1+13.4=103.3 seconds
# Schmekerezada (compiled with CLANG) is only 103.3/79.2=1.30x or 30% faster than GNUsort 16threads, cold shower for those who (like me) underestimated Mergesort's parallelizability.
```

#	Corpus Name	linux-6.1.38.tar	Human_Genome_DNA	30000000.KnightTours.txt	Fedora-Workstat...-38-1.6.iso
#	Corpus Size in Bytes	1,359,441,920	3,313,061,631	3,870,000,000	2,099,451,904
#	Sorter \ Corpus Size in Lines	35,585,653	40,902,071	30,000,000	8,189,810
#	sort v9.1 --parallel=4 -T ./	102,796,467,972 instructions 21,511,871,163 branches 451,366,228 branch-misses 38.9 seconds	145,940,013,705 instructions 30,368,703,735 branches 646,691,572 branch-misses 89.6 seconds	134,526,463,714 instructions 27,618,897,382 branches 248,678,541 branch-misses 101.2 seconds	32,039,863,819 instructions 6,671,955,880 branches 182,670,430 branch-misses 22.6 seconds
#	sort v9.1 --parallel=16 -T ./	95,974,435,959 instructions 20,010,961,812 branches 457,525,469 branch-misses 34.6 seconds	139,181,029,596 instructions 28,921,319,248 branches 638,439,543 branch-misses 76.0 seconds	121,021,375,460 instructions 24,752,255,755 branches 252,280,650 branch-misses 97.7 seconds	43,584,980,865 instructions 9,001,685,334 branches 251,235,618 branch-misses 31.6 seconds
#	Schmekerezada_CLANG_16.0.1 (v19)	65,342,898,585 instructions 13,854,546,733 branches 442,781,547 branch-misses 33.1 seconds	106,570,792,202 instructions 22,916,875,802 branches 565,776,851 branch-misses 61.4 seconds	137,387,012,432 instructions 31,582,212,549 branches 518,222,887 branch-misses 50.7 seconds	38,298,101,683 instructions 9,150,687,078 branches 125,573,907 branch-misses 20.9 seconds
#	Schmekerezada_GCC_13.0.1 (v19)	75,034,087,501 instructions 14,933,728,465 branches 436,610,153 branch-misses 35.2 seconds	123,284,759,010 instructions 25,022,793,479 branches 566,297,480 branch-misses 65.4 seconds	150,447,993,706 instructions 33,785,964,900 branches 545,784,186 branch-misses 52.7 seconds	43,962,088,192 instructions 9,980,116,461 branches 127,345,641 branch-misses 23.3 seconds

```
# Note1: The benchmark is in 'performance' mode as superuser;
```

```
# Note2: Linux version 6.2.12-300.fc38.x86_64 (mockbuild@54604edad16f4e818e702bda973f7473) (gcc (GCC) 13.0.1 20230401 (Red Hat 13.0.1-0), GNU ld version 2.39-9.fc38) #1 SMP PREEMPT_DYNAMIC
Thu Apr 20 23:05:25 UTC 2023
# Note3: Testmachine: Laptop Thinkpad 11e, Celeron N4100 CPU, 2.4GHz max turbo 4cores/4threads, L1d: 96 KiB (4 instances), L1i: 128 KiB (4 instances), L2: 4 MiB (1 instance), 8GB DDR4
2400MT/s, running Fedora 38;
# Note4: Schmekerezada is tetrathreaded;
# Note5: The current drive: SSD nvme 1TB TS1TMTE400S (DRAM-less cache);
# Note6a: Partition in use:
# Note6b: Filesystem      Type  Size  Used Avail Use% Mounted on
# Note6c: /dev/nvme0n1p2 ext4  875G  567G  264G   69% /
# Note7: LC_ALL=C locale was used for sort;
# Note8: After sorting, checking the shasum for both outputs - they all matched;
# Note9: CLANG compiler is significantly better than GCC, too many times;
# NoteA: The KT_30M corpus is of fixed-line size - 128 bytes each line - all lines unique, here the LittleEndian-To-BigEndian technique pays off;
# NoteB: The time statistics (wall clock) are reported by Linux' perf;
# NoteC: It is worth mentioning that CLANG executable is executed before GCC counterpart, it means possible caching of the whole file is more likely for the latter.
#
# So, cumulatively:
# 33.1+61.4+50.7+20.9=166.1 seconds
# 34.6+76.0+97.7+31.6=239.9 seconds
# Schmekerezada (compiled with CLANG) is only 239.9/166.1=1.44x or 44% faster than GNUsort 16threads, cold shower for those who (like me) underestimated Mergesort's parallelizability.
```

bench_PARAMETER.sh:

```
#sudo mkdir /tmp/ramdisk
#sudo chmod 777 /tmp/ramdisk
#sudo mount -t tmpfs -o size=32G myramdisk /tmp/ramdisk
#sudo umount /tmp/ramdisk/

cat /proc/version
echo
sudo fdisk -l
echo
lsblk -o NAME,FSTYPE,SIZE,MOUNTPOINT,MODEL,LABEL
echo
df -h -T
echo
echo Partition in use:
df -hT .
echo
echo performance | tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
echo
lscpu
echo
free -h
echo
#[root@kaze Schmekerezada]# swapon -a
#[root@kaze Schmekerezada]# swapon --show
```

```

#NAME      TYPE      SIZE USED PRIO
#/dev/sdb2 partition 128.8G  0B   -2
#[root@kaze Schmekerezada]# swapoff -a
#[root@kaze Schmekerezada]# swapon --show
#[root@kaze Schmekerezada]#

echo Enabling swap...
swapon -a
echo Swap status:
swapon --show
echo
export LC_ALL='C'
locale
echo $LC_ALL

cp "$1" /dev/null
date +%T
perf stat -d ./Schmekerezada_CLANG_16.0.1_SSE4.2_TetraThread.elf "$1"
shasum Schmekeriada.txt
perf stat -d ./Schmekerezada_GCC_13.0.1_SSE4.2_TetraThread.elf "$1"
date +%T
#/bin/time -v ./Schmekeriada_GCC_12.1.1_TetraThread.elf "$1"
sort --version
date +%T
perf stat -d sort -o Linuxsort "$1" --parallel=4 -T ./
date +%T
#/bin/time -v sort -o Linuxsort "$1" --parallel=4 -T ./
shasum Schmekeriada.txt
shasum Linuxsort
date +%T
perf stat -d sort -o Linuxsort "$1" --parallel=16 -T ./
date +%T
#/bin/time -v sort -o Linuxsort "$1" --parallel=16 -T ./
rm Schmekeriada.txt
rm Linuxsort

```