LEPRECHAUN

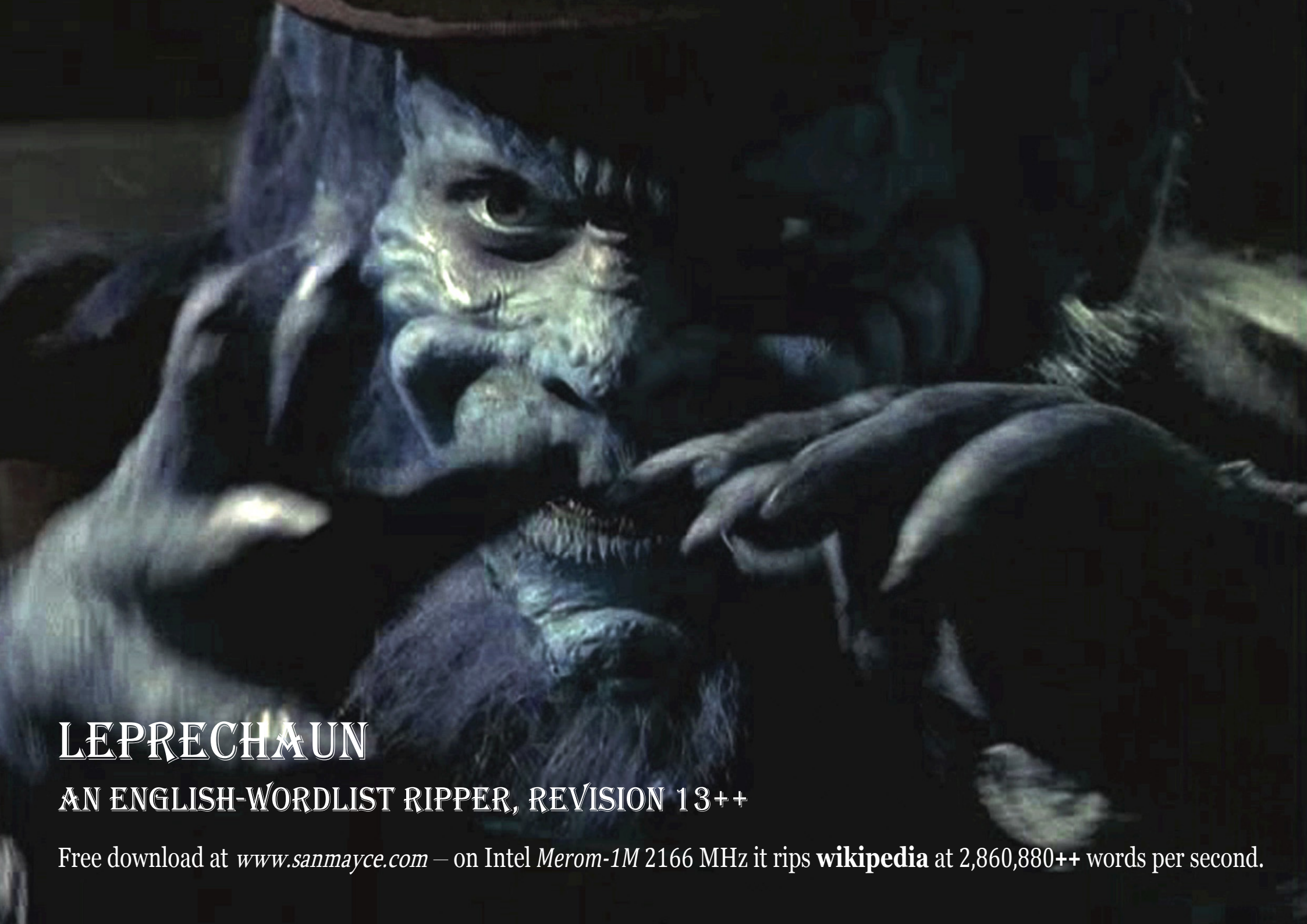
AN ENGLISH-WORDLIST RIPPER, REVISION 13++

Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.

LEPRECHAUN

AN ENGLISH-WORDLIST RIPPER, REVISION 13++

Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.



LEPRECHAUN

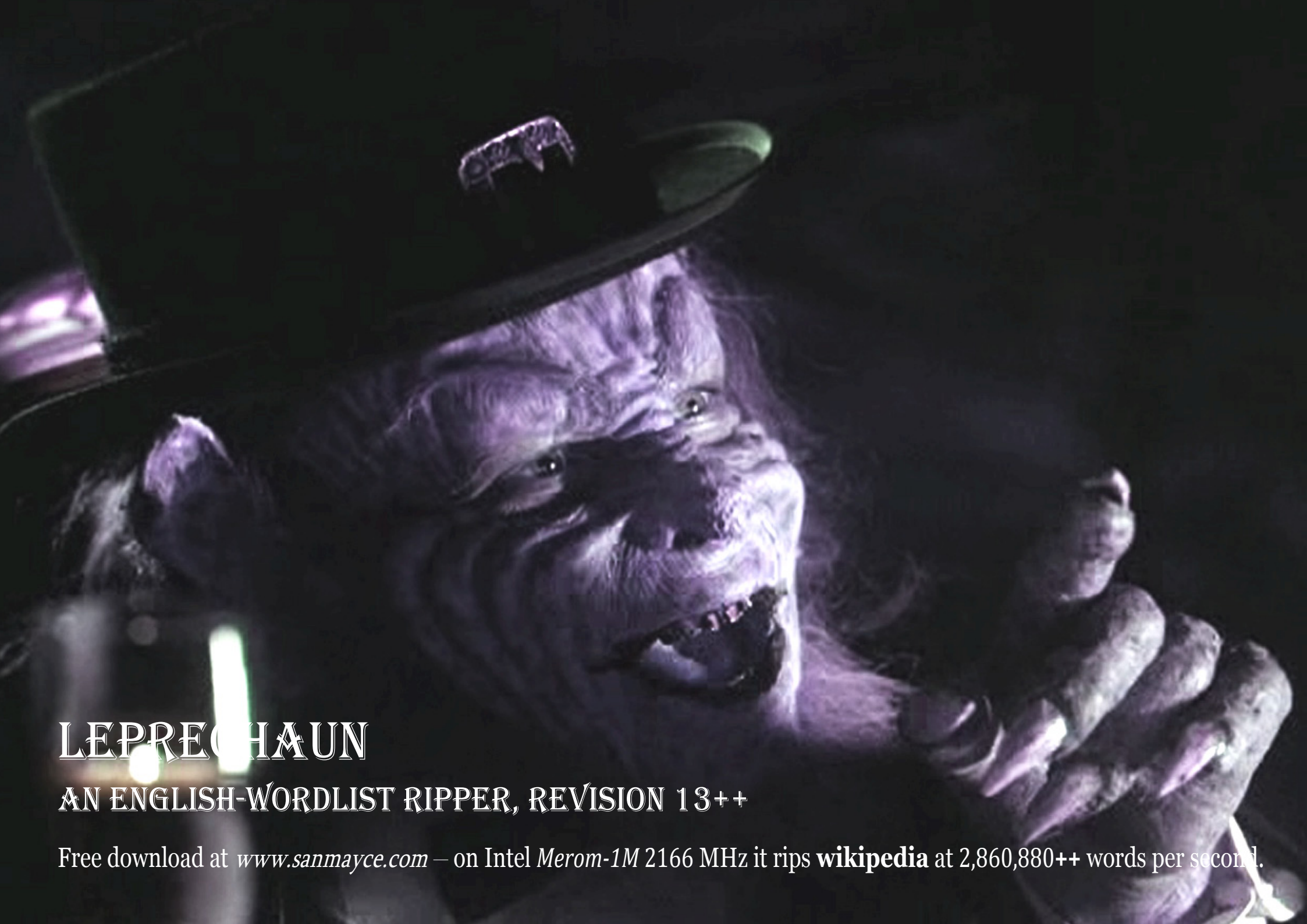
AN ENGLISH-WORDLIST RIPPER, REVISION 13++

Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.

LEPRECHAUN

AN ENGLISH-WORDLIST RIPPER, REVISION 13++

Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.



LEPRECHAUN

AN ENGLISH-WORDLIST RIPPER, REVISION 13++

Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.

LEPRECHAUN

AN ENGLISH-WORDLIST RIPPER, REVISION 13++

Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.


LEPRECHAUN
AN ENGLISH-WORDLIST RIPPER, REVISION 13++
Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.

LEPRECHAUN

AN ENGLISH-WORDLIST RIPPER, REVISION 13++

Free download at *www.sanmayce.com* – on Intel *Merom-1M* 2166 MHz it rips **wikipedia** at 2,860,880**++** words per second.

```
D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13++_ELF+EXE_vs_Wikipedia_22,202,980_LATIN-Words>dir

04/13/2010  05:56 AM           166,780 Leprechaun_r13++_32bits.asm
04/13/2010  05:56 AM            77,824 Leprechaun_r13++_32bits.exe
04/13/2010  05:56 AM           643,787 Leprechaun_r13++_generic_32bits.elf
04/13/2010  05:56 AM               132 Leprechaun_vs_Wikipedia_LATIN-WORDS.bat
04/13/2010  05:56 AM               216 Leprechaun_vs_Wikipedia_LATIN-WORDS.lst
04/13/2010  05:56 AM             1,635 Leprechaun_vs_Wikipedia_LATIN-WORDS.txt
04/13/2010  05:56 AM        98,215,517 wikipedia-de-html.tar.wrd
04/13/2010  05:56 AM       146,973,879 wikipedia-en-html.tar.wrd
04/13/2010  05:56 AM        31,913,244 wikipedia-es-html.tar.wrd
04/13/2010  05:56 AM        37,784,445 wikipedia-fr-html.tar.wrd
04/13/2010  05:56 AM        32,880,630 wikipedia-it-html.tar.wrd
04/13/2010  05:56 AM        34,311,298 wikipedia-nl-html.tar.wrd
04/13/2010  05:56 AM        23,830,432 wikipedia-pt-html.tar.wrd
04/13/2010  05:56 AM        10,073,451 wikipedia-ro-html.tar.wrd
              14 File(s)    416,873,270 bytes
               2 Dir(s)     896,507,904 bytes free

D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13++_ELF+EXE_vs_Wikipedia_22,202,980_LATIN-
Words>"Leprechaun_r13++_32bits.exe"
Leprechaun(Fast Greedy Word-Ripper), revision 13++, written by Svalqyatchx.
Leprechaun: 'Oh, well, didn't you hear? Bigger is good, but jumbo is dear.'
Kaze: Let's see what a 4-way hash + 6,602,752 Binary-Search-Trees can give us,
      also the performance of a 4-way hash + 6,602,752 B-Trees of order 3.

'The Little Monster' short notes:
Note1: I wish to thank to R.N. Horspool, Ranjan Sinha, Dmitry Shkarin,
       Michael Abrash, J. Bentley, R. Sedgewick, Igor Pavlov, Lasse Reinhold
       for sharing their knowledge to public.
Note2: Run it without parameters to get usage and short notes.
Note3: This simple amateurish(more over I am not versed well neither in C nor
       in mathematics nor in english language, but I am persistent in INDEXING
       GBs of english TEXTS) tool is written in ANSI C(at least its source is
       compileable for CL(Windows) and GCC(Linux)), and its purpose is to
       create a WordList for a group of files(given via filelist).
       Its name comes(according to Heritage Dictionary) from 'low corpus' or
```

'little body', in fact from amazing movie saga 'Leprechaun 1-2-3-4-5-6'
starring by Warwick Davis.
Note4: Only words up to 31 chars are proceeded - the reason is 'DDT'(the
longest word in Heritage Dictionary 3rd edition) or
'dichlorodiphenyltrichloroethane'.
Note5: Cursor hiding in C - mission impossible for me.
Note6: By default(third parameter is 1023) allocated memory is 393MB.
Due to 'malloc()' limitation under WINDOWS, maximum value of third
parameter is 5174 which is 1988MB allocated block.
Note7: File Leprechaun.LOG is a log, where new statistics are appended.
Note8: Revision 12+ can handle files larger than 4GB.
Note9: Revision 12++ has a buffered 'fread()' - therefore I/O READ-BURST SPEED
is the first(worst) bottleneck, as a result r.12++ is much-much faster;
the second(worse) bottleneck: the linked lists - the b-trees
might be the answer; the third(bad) bottleneck: the amateurish author.
NoteA: Revision 12+++ has an improved(2 bits were used doltishly) main hash
function - therefore less collisions, for example:
for file 'wikipedia-de-html.tar' 42,291,855,360 bytes with
5,750,179,678 words of them 7,375,373 distinct attempts to Find/Put
a WORD into a linked list are 6,117,675,470(r.12++) and 5,845,989,790
(r.12+++); also two 'if' sections were moved because they were executed
unnecessarily many times.
NoteB: Revision 13 uses BSTs instead of LLs, that is Linked-Lists were
replaced by Binary-Search-Trees, as a result for 22,202,980 distinct
words(out of 35,271,297) r.12+++ needs 225,548,268 total attempts to
Find/Put WORDs into linked lists where r.13 needs 121,674,042 total
attempts to Find/Put WORDs into Binary-Search-Trees. But this is a
significant boost in performance only for wordlists of million words.
NoteC: Revision 13+ gives only more statistics. Future revisions could lessen
number of attempts to Find/Put WORDs into Binary-Search-Trees
furthermore by making them at some point Perfectly-Balanced. But
for huge amount(multi-(m|b)illion) of distinct words the b-tree family
must come in, until then this is the leprechaunish niche.
NoteD: Revision 13++ has a little fix(2 unnecessary ZEROings, when a new word
is inserted, were deleted) and a fixed bug(13+ adds stupidly the
highest BST to the wordlist). Also B-Tree of order 3 is added as a
searching method. Main goal of B-Tree is to reduce number of

```
       comparisons but at nasty cost: a precious time wasted to construct it
       and twice more memory, i.e. one step forward two backward: this tree is
       more effective than BST in cases of 2++ billion/million
       different/distinct words.
       The improvement which comes from using B-Tree of order 3 is about 200%
       much more pleasing than I expected, for wikipedia-en-html.tar.wrd with
       12,561,874 distinct words Total Attempts to Find/Put WORDs into:
       Binary-Search-Trees was 61,895,043 while for
       B-trees order 3 was 19,295,791.
NoteE: For old r.12+ a USB connected HDD crippled test:
       for 'H:\>Leprechaun.exe static.wikipedia.org_downloads_2008-06_en.lst
       wikipedia-en-html.tar.wrd 5400'
       where 223,674,511,360 wikipedia-en-html.tar
       on laptop Toshiba Pentium T3400 2166 MHz with
       Motherboard Name:                                Toshiba Satellite L305
       CPU Type:         Mobile DualCore Intel Pentium, 2166 MHz (13 x 167)
       CPU Alias:                                                    Merom-1M
       L1 Code Cache:                                          32 KB per core
       L1 Data Cache:                                          32 KB per core
       L2 Cache:                        1 MB (On-Die, ECC, ASC, Full-Speed)
       Bus Type:                                             Dual DDR2 SDRAM
       Bus Width:                                                    128-bit
       Real Clock:                                            333 MHz (DDR)
       Effective Clock:                                              666 MHz
       EVEREST v5.00.1650 Memory Copy:       3725MB/s with timings 5-5-5-13
       result is logged to 'Leprechaun.LOG':
  Bytes per second performance: 20,658,955B/s
  Words per second performance: 2,860,880W/s
  Input File with a list of TEXTual Files:
   static.wikipedia.org_downloads_2008-06_en.lst
  Size of all TEXTual Files: 223,674,511,360
  Word count: 30,974,750,142 of them 12,561,874 distinct
  Number Of Files: 1
  Number Of Lines: 2088618575
  Allocated memory in MB: 1920
  Words with length 01 occupy 0,033KB of 0,349KB given i.e. 09% utilization
  Words with length 02 occupy 0,033KB of 0,349KB given i.e. 09% utilization
```

```
  Words with length 03 occupy 0,037KB of 0,697KB given i.e. 05% utilization
  Words with length 04 occupy 0,151KB of 0,871KB given i.e. 17% utilization
  Words with length 05 occupy 0,744KB of 1,568KB given i.e. 47% utilization
  Words with length 06 occupy 1,470KB of 3,136KB given i.e. 46% utilization
  Words with length 07 occupy 2,605KB of 5,923KB given i.e. 43% utilization
  Words with length 08 occupy 3,296KB of 6,968KB given i.e. 47% utilization
  Words with length 09 occupy 3,714KB of 6,968KB given i.e. 53% utilization
  Words with length 10 occupy 3,483KB of 6,968KB given i.e. 49% utilization
  Words with length 11 occupy 3,235KB of 5,923KB given i.e. 54% utilization
  Words with length 12 occupy 2,691KB of 4,181KB given i.e. 64% utilization
  Words with length 13 occupy 2,230KB of 3,484KB given i.e. 64% utilization
  Words with length 14 occupy 1,718KB of 3,484KB given i.e. 49% utilization
  Words with length 15 occupy 1,357KB of 2,613KB given i.e. 51% utilization
  Words with length 16 occupy 1,063KB of 2,613KB given i.e. 40% utilization
  Words with length 17 occupy 0,814KB of 1,742KB given i.e. 46% utilization
  Words with length 18 occupy 0,617KB of 1,742KB given i.e. 35% utilization
  Words with length 19 occupy 0,485KB of 1,742KB given i.e. 27% utilization
  Words with length 20 occupy 0,402KB of 1,742KB given i.e. 23% utilization
  Words with length 21 occupy 0,327KB of 1,742KB given i.e. 18% utilization
  Words with length 22 occupy 0,274KB of 1,742KB given i.e. 15% utilization
  Words with length 23 occupy 0,224KB of 1,394KB given i.e. 16% utilization
  Words with length 24 occupy 0,190KB of 1,394KB given i.e. 13% utilization
  Words with length 25 occupy 0,162KB of 1,394KB given i.e. 11% utilization
  Words with length 26 occupy 0,136KB of 1,220KB given i.e. 11% utilization
  Words with length 27 occupy 0,119KB of 1,046KB given i.e. 11% utilization
  Words with length 28 occupy 0,107KB of 0,871KB given i.e. 12% utilization
  Words with length 29 occupy 0,091KB of 0,697KB given i.e. 13% utilization
  Words with length 30 occupy 0,080KB of 0,523KB given i.e. 15% utilization
  Words with length 31 occupy 0,076KB of 0,523KB given i.e. 14% utilization
  Total pseudo(including hash table) memory utilization: 42%
  Total real(wordlist's words VS allocated block) memory utilization: 60/1000
  Used value for third parameter in KB: 5400
  Use next time as third parameter: 3475-
  Time for making unsorted wordlist: 10827 second(s)
  Time for sorting unsorted wordlist: 10 second(s)

Usage: Leprechaun InFile OutFile [BufferSize] [SortMethod] [TreeMethod]
```

```
        <InFile>: Input file with files for Leprechauning, in WINDOWS console
                  you can create it by 'E:\KAZEHOME>dir *.txt/s/b>Leprechaun.lst'
        <OutFile>: Output WORDLIST(sorted since r.9, CRLF) file
        <BufferSize>: Optional Dynamic RAM buffer in KB, default(and minimum
                      in the same time) is 1023, i.e. omit or specify greater one
        <SortMethod>: Optional Sort Method, default is 'D',
                      A - InsertionSort
                      B - InsertionX26Sort
                      C - MultiKeyQuickSortSort by J. Bentley, R. Sedgewick
                      D - MultiKeyQuickSortX26Sort' by J. Bentley, R. Sedgewick
        <TreeMethod>: Optional Tree Method, default is 'X',
                      X - Binary-Search-Trees
                      Y - B-Trees of order 3

Have a nice Leprechauning.
For contacts: sanmayce@hotmail.com
Sanmayce Svalqyatchx 'Kaze', 2005 Feb 07(rev.13++: 2010 Apr 12).

D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13++_ELF+EXE_vs_Wikipedia_22,202,980_LATIN-Words>type
Leprechaun_vs_Wikipedia_LATIN-WORDS.bat
@echo off
Leprechaun_r13++_32bits.exe Leprechaun_vs_Wikipedia_LATIN-WORDS.lst Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd 5000
echo.

D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13++_ELF+EXE_vs_Wikipedia_22,202,980_LATIN-
Words>Leprechaun_vs_Wikipedia_LATIN-WORDS.bat
Leprechaun(Fast Greedy Word-Ripper), revision 13++, written by Svalqyatchx.
Leprechaun: 'Oh, well, didn't you hear? Bigger is good, but jumbo is dear.'
Kaze: Let's see what a 4-way hash + 6,602,752 Binary-Search-Trees can give us,
      also the performance of a 4-way hash + 6,602,752 B-Trees of order 3.
Size of input file with files for Leprechauning: 216
Allocated memory in MB: 1950
Size of Input TEXTual file: 98,215,517
|; Word count: 7,375,373 of them 7,375,373 distinct; Done: 64/64
Size of Input TEXTual file: 146,973,879
/; Word count: 19,937,247 of them 17,322,675 distinct; Done: 64/64
Size of Input TEXTual file: 31,913,244
```

```
|; Word count: 22,829,701 of them 18,291,299 distinct; Done: 64/64
Size of Input TEXTual file: 37,784,445
/; Word count: 26,296,387 of them 19,346,269 distinct; Done: 64/64
Size of Input TEXTual file: 32,880,630
|; Word count: 29,256,183 of them 20,331,005 distinct; Done: 64/64
Size of Input TEXTual file: 34,311,298
|; Word count: 32,128,367 of them 21,393,001 distinct; Done: 64/64
Size of Input TEXTual file: 23,830,432
\; Word count: 34,310,324 of them 21,978,966 distinct; Done: 64/64
Size of Input TEXTual file: 10,073,451
/; Word count: 35,271,297 of them 22,202,980 distinct; Done: 64/64
Flushing unsorted words ...
Time for making unsorted wordlist: 45 second(s)
Deallocated memory in MB: 1950
Allocated memory for words in MB: 266
Allocated memory for pointers-to-words in MB: 85
Sorting(with 'MultiKeyQuickSortX26Sort' by J. Bentley and R. Sedgewick) ...
Sort pass 26/26 ...
Flushing sorted words ...
Time for sorting unsorted wordlist: 17 second(s)
Leprechaun: Done.

D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13++_ELF+EXE_vs_Wikipedia_22,202,980_LATIN-Words>type
Leprechaun.LOG
Leprechaun report:
A(not always THE) Binary-Search-Tree with the longest path(height, PEAK, numberof levels):
                                      ]ezulueta]
                                       [ewexipta]
                                     [eufopoli[
                                    [eturtleh[
                                   [etselkmn[
                                  [ethnonat[
                                 [ethianos[
                                [esnowmoa[
                               [eskesoni[
                              [esinwell[
                             [eshupark[
```

[esheloby[
[eshaukom[
]esfkopin[
[escrerve]
[escepter[
[erlkings[
]ereaxion[
[epompoen]
]epathysa[
[epariolo]
[eopowiem[
[enitrome[
[emyoaung[
]emontram[
[emititdo]
]emicaiah[
[emassoli]
[emarkydn[
]emajnoon[
]elvismen]
[elvgulik]
[elishawn[
]elincuri[
[ekwigybo]
[ekroatoj[
]ekhaosay[
[ekelvish]
[ejoutman[
[ejonjper[
]ejbenett[
[eijhusen]
]eigendst[
[ehopmans]
[ehefelea[
]egothicx[
[egoeroes]
[egdevils[

```
      [eflagrum[
     ]edopisni[
      [ederekto]
    [edelsens[
   ]eckzahns[
    [eccarton]
  [ecarboot[
 ]ebugtraq[ ROOT
  ]ebeetley]
   [eahivyle]
Above Binary-Search-Tree with MaxPEAK = 38 has NODEs = 58 and LEAFs = 15
Legend:
At left side of the word - '[' means no left successor
At left side of the word - ']' means left successor exists
At right side of the word - ']' means no right successor
At right side of the word - '[' means right successor exists
Bytes per second performance: 9,244,064B/s
Words per second performance: 783,806W/s
Input File with a list of TEXTual Files: Leprechaun_vs_Wikipedia_LATIN-WORDS.lst
Size of all TEXTual Files: 415,982,896
Word count: 35,271,297 of them 22,202,980 distinct
Number Of Files: 8
Number Of Lines: 35271297
Allocated memory in MB: 1950
Number Of Trees(GREATER THE BETTER): 3410463
Forest population(Hash Function Quality regarding Collisions i.e. Hash Table Utilization): 51%
Number Of Hash Collisions(Distinct WORDs - Number Of Trees): 18792517
Maximum Attempts to Find/Put a WORD into a Binary-Search-Tree: '38'
Total Attempts to Find/Put WORDs into Binary-Search-Trees: 121,674,042
Total Number of LEAFs in Binary-Search-Trees(GREATER THE BETTER): 7,990,635
Perfectly-Balanced-Binary-Search-Tree for MaxNODEs = 94 must have PEAK = 7 = rounding down of integer (1+lb(94))
Binary-Search-Tree(1st out of 1) with MaxNODEs = 94 has PEAK = 20 and LEAFs = 29
Binary-Search-Tree(1st out of 2) with MaxPEAK = '38' has NODEs = 58 and LEAFs =15
Binary-Search-Tree(1st out of 2) with MaxLEAFs = 30 has NODEs = 93 and PEAK = 23
Words with length 01 occupy 0,033KB of 0,162KB given i.e. 19% utilization
Words with length 02 occupy 0,033KB of 0,162KB given i.e. 19% utilization
Words with length 03 occupy 0,040KB of 0,162KB given i.e. 24% utilization
```

```
Words with length 04 occupy 0,224KB of 0,646KB given i.e. 34% utilization
Words with length 05 occupy 1,311KB of 1,775KB given i.e. 73% utilization
Words with length 06 occupy 2,902KB of 3,549KB given i.e. 81% utilization
Words with length 07 occupy 5,345KB of 5,968KB given i.e. 89% utilization
Words with length 08 occupy 6,826KB of 7,581KB given i.e. 90% utilization
Words with length 09 occupy 7,683KB of 8,549KB given i.e. 89% utilization
Words with length 10 occupy 7,193KB of 8,065KB given i.e. 89% utilization
Words with length 11 occupy 6,606KB of 7,420KB given i.e. 89% utilization
Words with length 12 occupy 5,514KB of 6,130KB given i.e. 89% utilization
Words with length 13 occupy 4,599KB of 5,162KB given i.e. 89% utilization
Words with length 14 occupy 3,636KB of 4,033KB given i.e. 90% utilization
Words with length 15 occupy 2,900KB of 3,226KB given i.e. 89% utilization
Words with length 16 occupy 2,286KB of 2,904KB given i.e. 78% utilization
Words with length 17 occupy 1,763KB of 2,259KB given i.e. 78% utilization
Words with length 18 occupy 1,355KB of 1,613KB given i.e. 83% utilization
Words with length 19 occupy 1,065KB of 1,291KB given i.e. 82% utilization
Words with length 20 occupy 0,843KB of 1,130KB given i.e. 74% utilization
Words with length 21 occupy 0,659KB of 0,968KB given i.e. 68% utilization
Words with length 22 occupy 0,530KB of 0,807KB given i.e. 65% utilization
Words with length 23 occupy 0,418KB of 0,646KB given i.e. 64% utilization
Words with length 24 occupy 0,337KB of 0,484KB given i.e. 69% utilization
Words with length 25 occupy 0,278KB of 0,484KB given i.e. 57% utilization
Words with length 26 occupy 0,223KB of 0,323KB given i.e. 68% utilization
Words with length 27 occupy 0,182KB of 0,323KB given i.e. 56% utilization
Words with length 28 occupy 0,161KB of 0,323KB given i.e. 49% utilization
Words with length 29 occupy 0,131KB of 0,323KB given i.e. 40% utilization
Words with length 30 occupy 0,111KB of 0,162KB given i.e. 68% utilization
Words with length 31 occupy 0,100KB of 0,162KB given i.e. 61% utilization
Total pseudo(including hash table) memory utilization: 85%
Total real(wordlist's words VS allocated block) memory utilization: 114/1000
Used value for third parameter in KB: 5000
Use next time as third parameter: 4509-
Time for making unsorted wordlist: 45 second(s)
Time for sorting unsorted wordlist: 17 second(s)

D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13++_ELF+EXE_vs_Wikipedia_22,202,980_LATIN-Words>
```

D:\Leprechaun_vs_Wikipedia_22,202,980_LATIN-Words>**type Leprechaun_vs_Wikipedia_LATIN-WORDS.txt**
Wikipedia Static HTML Dumps_static.wikipedia.org_downloads_2008-06:

en:
Word count: 30,974,750,142 of them 12,561,874 distinct
wikipedia-en-html.tar 223,674,511,360
wikipedia-en-html.tar.7z 15,363,543,213
wikipedia-en-html.tar.wrd 146,973,879

de:
Word count: 5,750,179,678 of them 7,375,373 distinct
wikipedia-de-html.tar 42,291,855,360
wikipedia-de-html.tar.7z 3,389,371,118
wikipedia-de-html.tar.wrd 98,215,517

fr:
Word count: 6,097,411,556 of them 3,466,686 distinct
wikipedia-fr-html.tar 42,766,336,000
wikipedia-fr-html.tar.7z 2,569,418,524
wikipedia-fr-html.tar.wrd 37,784,445

ro:
Word count: 672,050,402 of them 960,973 distinct
wikipedia-ro-html.tar 4,891,637,760
wikipedia-ro-html.tar.7z 232,144,034
wikipedia-ro-html.tar.wrd 10,073,451

es:
Word count: 2,880,656,861 of them 2,892,454 distinct
wikipedia-es-html.tar 20,276,602,880
wikipedia-es-html.tar.7z 1,345,346,047
wikipedia-es-html.tar.wrd 31,913,244

it:
Word count: 3,860,030,144 of them 2,959,796 distinct
wikipedia-it-html.tar 27,932,119,040
wikipedia-it-html.tar.7z 1,743,914,079
wikipedia-it-html.tar.wrd 32,880,630

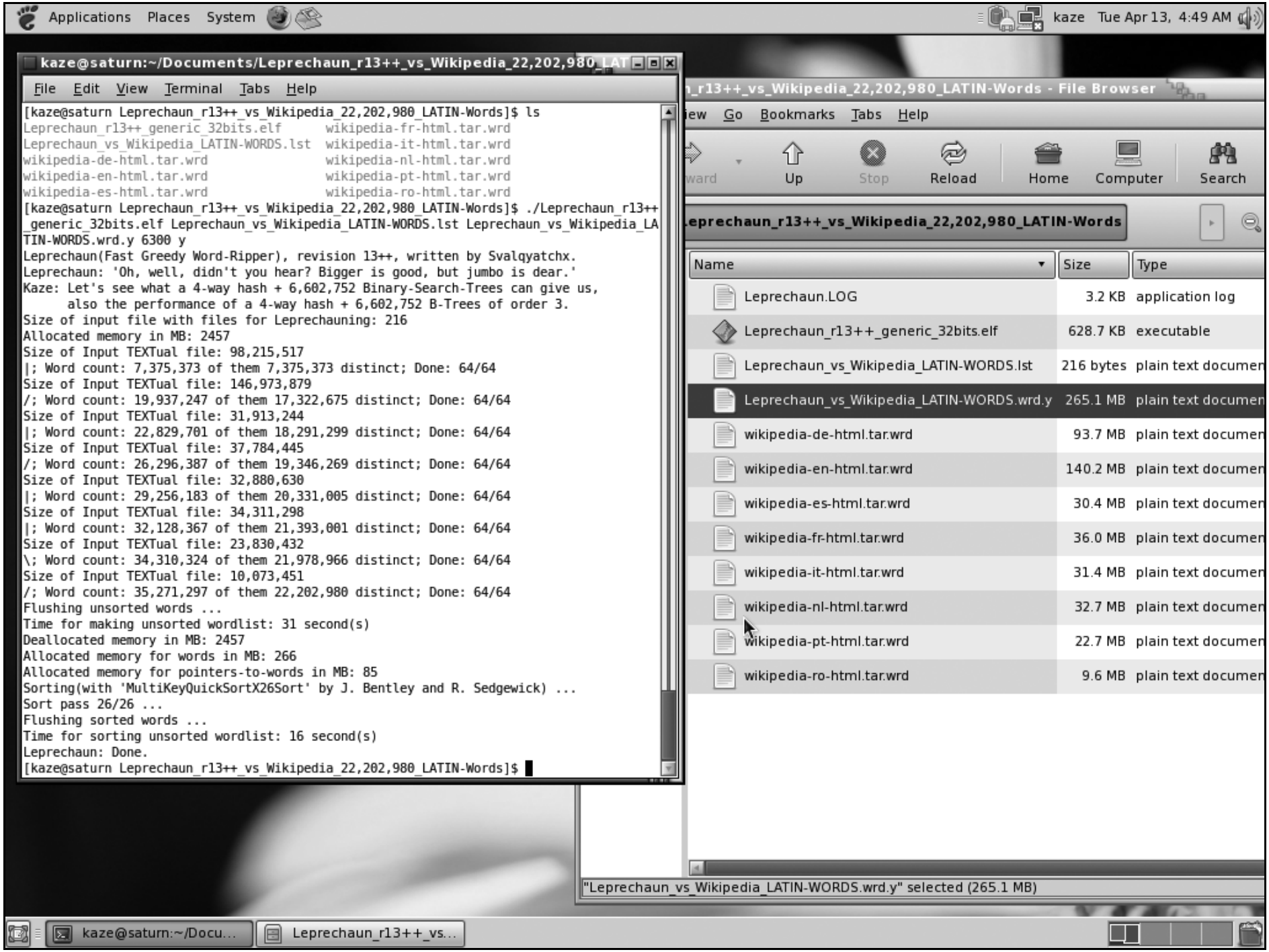
nl:
Word count: 2,678,680,521 of them 2,872,184 distinct
wikipedia-nl-html.tar 19,808,522,240
wikipedia-nl-html.tar.7z 1,079,963,039
wikipedia-nl-html.tar.wrd 34,311,298

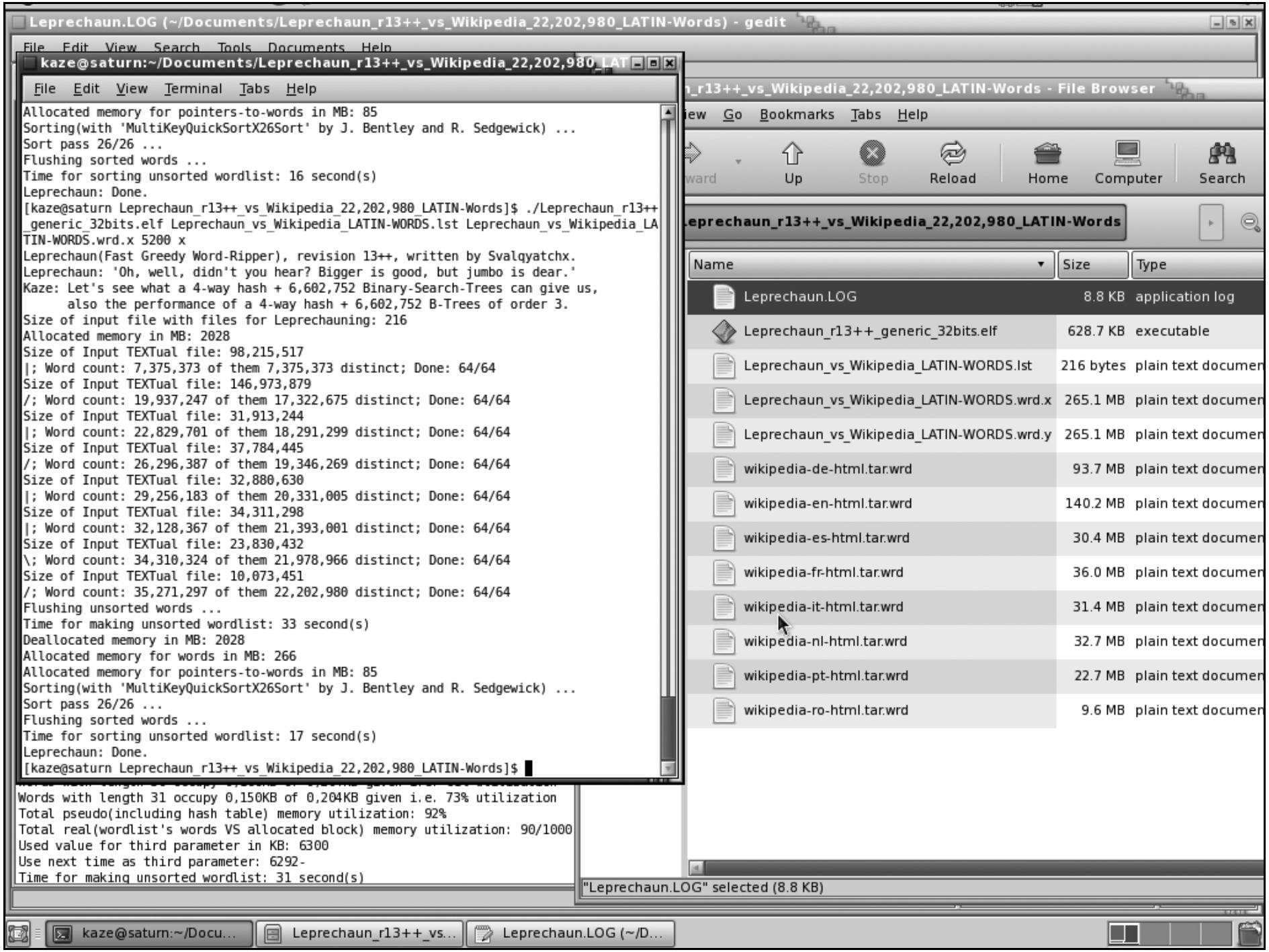
pt:
Word count: 3,342,724,894 of them 2,181,957 distinct
wikipedia-pt-html.tar 23,395,072,000
wikipedia-pt-html.tar.7z 955,302,104
wikipedia-pt-html.tar.wrd 23,830,432

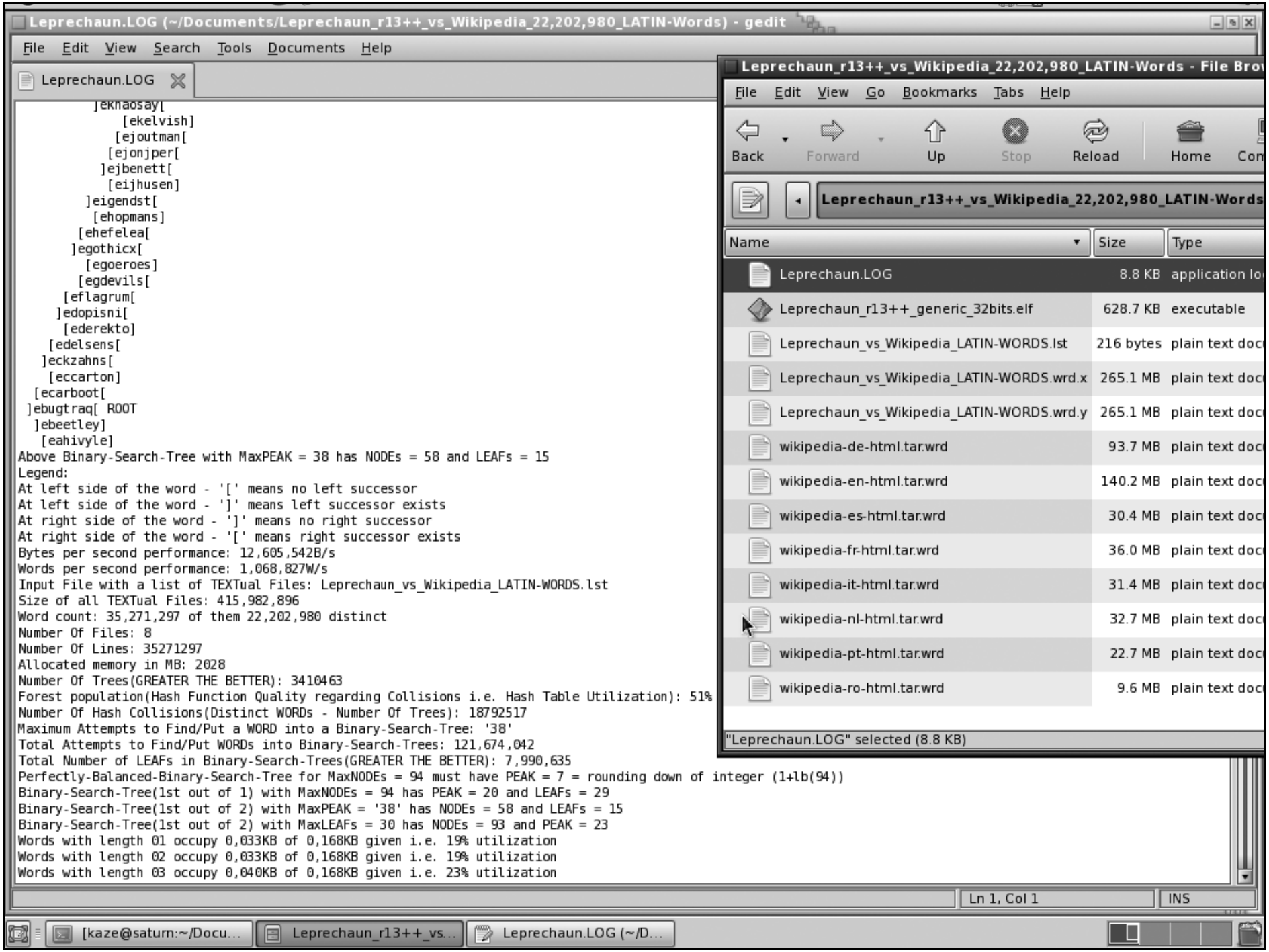
End.

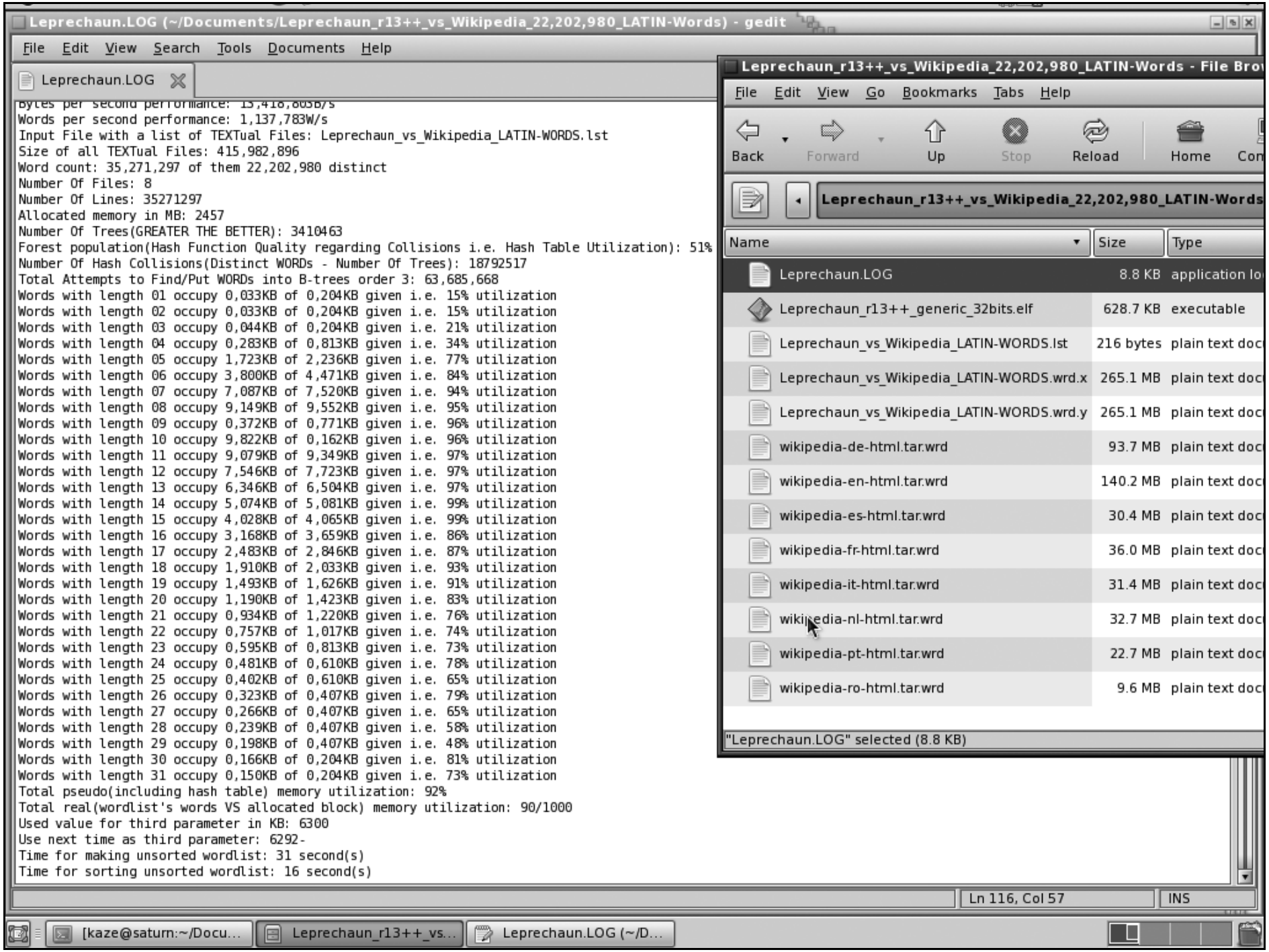
Leprechaun

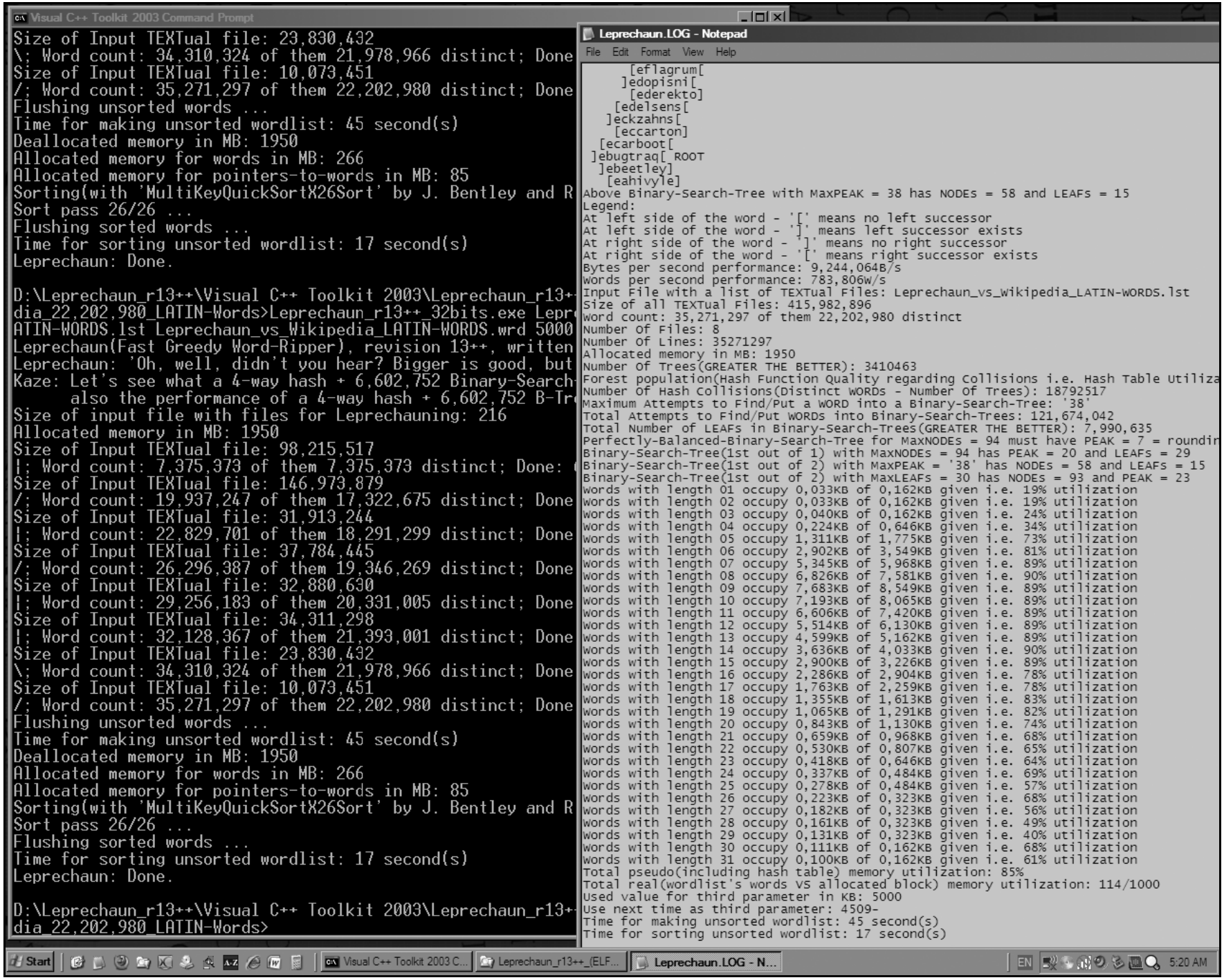
a word-list ripper with superior performance


Applications Places System
kaze Tue Apr 13, 4:49 AM
kaze@saturn:~/Documents/Leprechaun_r13++_vs_Wikipedia_22,202,980_LAT
File Edit View Terminal Tabs Help
[kaze@saturn Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words]$ ls
Leprechaun_r13++_generic_32bits.elf wikipedia-fr-html.tar.wrd
Leprechaun_vs_Wikipedia_LATIN-WORDS.lst wikipedia-it-html.tar.wrd
wikipedia-de-html.tar.wrd wikipedia-nl-html.tar.wrd
wikipedia-en-html.tar.wrd wikipedia-pt-html.tar.wrd
wikipedia-es-html.tar.wrd wikipedia-ro-html.tar.wrd
[kaze@saturn Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words]$ ./Leprechaun_r13++_generic_32bits.elf Leprechaun_vs_Wikipedia_LATIN-WORDS.lst Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.y 6300 y
Leprechaun(Fast Greedy Word-Ripper), revision 13++, written by Svalqyatchx.
Leprechaun: 'Oh, well, didn't you hear? Bigger is good, but jumbo is dear.'
Kaze: Let's see what a 4-way hash + 6,602,752 Binary-Search-Trees can give us,
also the performance of a 4-way hash + 6,602,752 B-Trees of order 3.
Size of input file with files for Leprechauning: 216
Allocated memory in MB: 2457
Size of Input TEXTual file: 98,215,517
|; Word count: 7,375,373 of them 7,375,373 distinct; Done: 64/64
Size of Input TEXTual file: 146,973,879
/; Word count: 19,937,247 of them 17,322,675 distinct; Done: 64/64
Size of Input TEXTual file: 31,913,244
|; Word count: 22,829,701 of them 18,291,299 distinct; Done: 64/64
Size of Input TEXTual file: 37,784,445
/; Word count: 26,296,387 of them 19,346,269 distinct; Done: 64/64
Size of Input TEXTual file: 32,880,630
|; Word count: 29,256,183 of them 20,331,005 distinct; Done: 64/64
Size of Input TEXTual file: 34,311,298
|; Word count: 32,128,367 of them 21,393,001 distinct; Done: 64/64
Size of Input TEXTual file: 23,830,432
\; Word count: 34,310,324 of them 21,978,966 distinct; Done: 64/64
Size of Input TEXTual file: 10,073,451
/; Word count: 35,271,297 of them 22,202,980 distinct; Done: 64/64
Flushing unsorted words ...
Time for making unsorted wordlist: 31 second(s)
Deallocated memory in MB: 2457
Allocated memory for words in MB: 266
Allocated memory for pointers-to-words in MB: 85
Sorting(with 'MultiKeyQuickSortX26Sort' by J. Bentley and R. Sedgewick) ...
Sort pass 26/26 ...
Flushing sorted words ...
Time for sorting unsorted wordlist: 16 second(s)
Leprechaun: Done.
[kaze@saturn Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words]$
r13++_vs_Wikipedia_22,202,980_LATIN-Words - File Browser
iew Go Bookmarks Tabs Help
ward Up Stop Reload Home Computer Search
eprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words
Name Size Type
Leprechaun.LOG 3.2 KB application log
Leprechaun_r13++_generic_32bits.elf 628.7 KB executable
Leprechaun_vs_Wikipedia_LATIN-WORDS.lst 216 bytes plain text documen
Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.y 265.1 MB plain text documen
wikipedia-de-html.tar.wrd 93.7 MB plain text documen
wikipedia-en-html.tar.wrd 140.2 MB plain text documen
wikipedia-es-html.tar.wrd 30.4 MB plain text documen
wikipedia-fr-html.tar.wrd 36.0 MB plain text documen
wikipedia-it-html.tar.wrd 31.4 MB plain text documen
wikipedia-nl-html.tar.wrd 32.7 MB plain text documen
wikipedia-pt-html.tar.wrd 22.7 MB plain text documen
wikipedia-ro-html.tar.wrd 9.6 MB plain text documen
"Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.y" selected (265.1 MB)
kaze@saturn:~/Docu...
Leprechaun_r13++_vs...


Leprechaun.LOG (~/Documents/Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words) - gedit
File Edit View Search Tools Documents Help
kaze@saturn:~/Documents/Leprechaun_r13++_vs_Wikipedia_22,202,980_LAT
File Edit View Terminal Tabs Help
Allocated memory for pointers-to-words in MB: 85
Sorting(with 'MultiKeyQuickSortX26Sort' by J. Bentley and R. Sedgewick) ...
Sort pass 26/26 ...
Flushing sorted words ...
Time for sorting unsorted wordlist: 16 second(s)
Leprechaun: Done.
[kaze@saturn Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words]$ ./Leprechaun_r13++_generic_32bits.elf Leprechaun_vs_Wikipedia_LATIN-WORDS.lst Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.x 5200 x
Leprechaun(Fast Greedy Word-Ripper), revision 13++, written by Svalqyatchx.
Leprechaun: 'Oh, well, didn't you hear? Bigger is good, but jumbo is dear.'
Kaze: Let's see what a 4-way hash + 6,602,752 Binary-Search-Trees can give us,
also the performance of a 4-way hash + 6,602,752 B-Trees of order 3.
Size of input file with files for Leprechauning: 216
Allocated memory in MB: 2028
Size of Input TEXTual file: 98,215,517
|; Word count: 7,375,373 of them 7,375,373 distinct; Done: 64/64
Size of Input TEXTual file: 146,973,879
/; Word count: 19,937,247 of them 17,322,675 distinct; Done: 64/64
Size of Input TEXTual file: 31,913,244
|; Word count: 22,829,701 of them 18,291,299 distinct; Done: 64/64
Size of Input TEXTual file: 37,784,445
/; Word count: 26,296,387 of them 19,346,269 distinct; Done: 64/64
Size of Input TEXTual file: 32,880,630
|; Word count: 29,256,183 of them 20,331,005 distinct; Done: 64/64
Size of Input TEXTual file: 34,311,298
|; Word count: 32,128,367 of them 21,393,001 distinct; Done: 64/64
Size of Input TEXTual file: 23,830,432
\; Word count: 34,310,324 of them 21,978,966 distinct; Done: 64/64
Size of Input TEXTual file: 10,073,451
/; Word count: 35,271,297 of them 22,202,980 distinct; Done: 64/64
Flushing unsorted words ...
Time for making unsorted wordlist: 33 second(s)
Deallocated memory in MB: 2028
Allocated memory for words in MB: 266
Allocated memory for pointers-to-words in MB: 85
Sorting(with 'MultiKeyQuickSortX26Sort' by J. Bentley and R. Sedgewick) ...
Sort pass 26/26 ...
Flushing sorted words ...
Time for sorting unsorted wordlist: 17 second(s)
Leprechaun: Done.
[kaze@saturn Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words]$
Words with length 31 occupy 0,150KB of 0,204KB given i.e. 73% utilization
Total pseudo(including hash table) memory utilization: 92%
Total real(wordlist's words VS allocated block) memory utilization: 90/1000
Used value for third parameter in KB: 6300
Use next time as third parameter: 6292-
Time for making unsorted wordlist: 31 second(s)
r13++_vs_Wikipedia_22,202,980_LATIN-Words - File Browser
iew Go Bookmarks Tabs Help
ward Up Stop Reload Home Computer Search
eprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words
Name Size Type
Leprechaun.LOG 8.8 KB application log
Leprechaun_r13++_generic_32bits.elf 628.7 KB executable
Leprechaun_vs_Wikipedia_LATIN-WORDS.lst 216 bytes plain text documen
Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.x 265.1 MB plain text documen
Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.y 265.1 MB plain text documen
wikipedia-de-html.tar.wrd 93.7 MB plain text documen
wikipedia-en-html.tar.wrd 140.2 MB plain text documen
wikipedia-es-html.tar.wrd 30.4 MB plain text documen
wikipedia-fr-html.tar.wrd 36.0 MB plain text documen
wikipedia-it-html.tar.wrd 31.4 MB plain text documen
wikipedia-nl-html.tar.wrd 32.7 MB plain text documen
wikipedia-pt-html.tar.wrd 22.7 MB plain text documen
wikipedia-ro-html.tar.wrd 9.6 MB plain text documen
"Leprechaun.LOG" selected (8.8 KB)
kaze@saturn:~/Docu...
Leprechaun_r13++_vs...
Leprechaun.LOG (~/D...


Leprechaun.LOG (~/Documents/Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words) - gedit
File Edit View Search Tools Documents Help
Leprechaun.LOG
]ekhaosay[
[ekelvish]
]ejoutman[
[ejonjper[
]ejbenett[
[eijhusen]
]eigendst[
[ehopmans]
[ehefelea[
]egothicx[
[egoeroes]
[egdevils[
[eflagrum[
]edopisni[
[ederekto]
[edelsens[
]eckzahns[
[eccarton]
[ecarboot[
]ebugtraq[ROOT
]ebeetley]
[eahivyle]
Above Binary-Search-Tree with MaxPEAK = 38 has NODEs = 58 and LEAFs = 15
Legend:
At left side of the word - '[' means no left successor
At left side of the word - ']' means left successor exists
At right side of the word - ']' means no right successor
At right side of the word - '[' means right successor exists
Bytes per second performance: 12,605,542B/s
Words per second performance: 1,068,827W/s
Input File with a list of TEXTual Files: Leprechaun_vs_Wikipedia_LATIN-WORDS.lst
Size of all TEXTual Files: 415,982,896
Word count: 35,271,297 of them 22,202,980 distinct
Number Of Files: 8
Number Of Lines: 35271297
Allocated memory in MB: 2028
Number Of Trees(GREATER THE BETTER): 3410463
Forest population(Hash Function Quality regarding Collisions i.e. Hash Table Utilization): 51%
Number Of Hash Collisions(Distinct WORDs - Number Of Trees): 18792517
Maximum Attempts to Find/Put a WORD into a Binary-Search-Tree: '38'
Total Attempts to Find/Put WORDs into Binary-Search-Trees: 121,674,042
Total Number of LEAFs in Binary-Search-Trees(GREATER THE BETTER): 7,990,635
Perfectly-Balanced-Binary-Search-Tree for MaxNODEs = 94 must have PEAK = 7 = rounding down of integer (1+lb(94))
Binary-Search-Tree(1st out of 1) with MaxNODEs = 94 has PEAK = 20 and LEAFs = 29
Binary-Search-Tree(1st out of 2) with MaxPEAK = '38' has NODEs = 58 and LEAFs = 15
Binary-Search-Tree(1st out of 2) with MaxLEAFs = 30 has NODEs = 93 and PEAK = 23
Words with length 01 occupy 0,033KB of 0,168KB given i.e. 19% utilization
Words with length 02 occupy 0,033KB of 0,168KB given i.e. 19% utilization
Words with length 03 occupy 0,040KB of 0,168KB given i.e. 23% utilization
Ln 1, Col 1
INS
Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words - File Bro
File Edit View Go Bookmarks Tabs Help
Back Forward Up Stop Reload Home
Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words
Name Size Type
Leprechaun.LOG 8.8 KB application lo
Leprechaun_r13++_generic_32bits.elf 628.7 KB executable
Leprechaun_vs_Wikipedia_LATIN-WORDS.lst 216 bytes plain text doc
Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.x 265.1 MB plain text doc
Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.y 265.1 MB plain text doc
wikipedia-de-html.tar.wrd 93.7 MB plain text doc
wikipedia-en-html.tar.wrd 140.2 MB plain text doc
wikipedia-es-html.tar.wrd 30.4 MB plain text doc
wikipedia-fr-html.tar.wrd 36.0 MB plain text doc
wikipedia-it-html.tar.wrd 31.4 MB plain text doc
wikipedia-nl-html.tar.wrd 32.7 MB plain text doc
wikipedia-pt-html.tar.wrd 22.7 MB plain text doc
wikipedia-ro-html.tar.wrd 9.6 MB plain text doc
"Leprechaun.LOG" selected (8.8 KB)
[kaze@saturn:~/Docu...
Leprechaun_r13++_vs...
Leprechaun.LOG (~/D...


Leprechaun.LOG (~/Documents/Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words) - gedit
File Edit View Search Tools Documents Help
Leprechaun.LOG
Bytes per second performance: 13,418,865B/s
Words per second performance: 1,137,783W/s
Input File with a list of TEXTual Files: Leprechaun_vs_Wikipedia_LATIN-WORDS.lst
Size of all TEXTual Files: 415,982,896
Word count: 35,271,297 of them 22,202,980 distinct
Number Of Files: 8
Number Of Lines: 35271297
Allocated memory in MB: 2457
Number Of Trees(GREATER THE BETTER): 3410463
Forest population(Hash Function Quality regarding Collisions i.e. Hash Table Utilization): 51%
Number Of Hash Collisions(Distinct WORDs - Number Of Trees): 18792517
Total Attempts to Find/Put WORDs into B-trees order 3: 63,685,668
Words with length 01 occupy 0,033KB of 0,204KB given i.e. 15% utilization
Words with length 02 occupy 0,033KB of 0,204KB given i.e. 15% utilization
Words with length 03 occupy 0,044KB of 0,204KB given i.e. 21% utilization
Words with length 04 occupy 0,283KB of 0,813KB given i.e. 34% utilization
Words with length 05 occupy 1,723KB of 2,236KB given i.e. 77% utilization
Words with length 06 occupy 3,800KB of 4,471KB given i.e. 84% utilization
Words with length 07 occupy 7,087KB of 7,520KB given i.e. 94% utilization
Words with length 08 occupy 9,149KB of 9,552KB given i.e. 95% utilization
Words with length 09 occupy 0,372KB of 0,771KB given i.e. 96% utilization
Words with length 10 occupy 9,822KB of 0,162KB given i.e. 96% utilization
Words with length 11 occupy 9,079KB of 9,349KB given i.e. 97% utilization
Words with length 12 occupy 7,546KB of 7,723KB given i.e. 97% utilization
Words with length 13 occupy 6,346KB of 6,504KB given i.e. 97% utilization
Words with length 14 occupy 5,074KB of 5,081KB given i.e. 99% utilization
Words with length 15 occupy 4,028KB of 4,065KB given i.e. 99% utilization
Words with length 16 occupy 3,168KB of 3,659KB given i.e. 86% utilization
Words with length 17 occupy 2,483KB of 2,846KB given i.e. 87% utilization
Words with length 18 occupy 1,910KB of 2,033KB given i.e. 93% utilization
Words with length 19 occupy 1,493KB of 1,626KB given i.e. 91% utilization
Words with length 20 occupy 1,190KB of 1,423KB given i.e. 83% utilization
Words with length 21 occupy 0,934KB of 1,220KB given i.e. 76% utilization
Words with length 22 occupy 0,757KB of 1,017KB given i.e. 74% utilization
Words with length 23 occupy 0,595KB of 0,813KB given i.e. 73% utilization
Words with length 24 occupy 0,481KB of 0,610KB given i.e. 78% utilization
Words with length 25 occupy 0,402KB of 0,610KB given i.e. 65% utilization
Words with length 26 occupy 0,323KB of 0,407KB given i.e. 79% utilization
Words with length 27 occupy 0,266KB of 0,407KB given i.e. 65% utilization
Words with length 28 occupy 0,239KB of 0,407KB given i.e. 58% utilization
Words with length 29 occupy 0,198KB of 0,407KB given i.e. 48% utilization
Words with length 30 occupy 0,166KB of 0,204KB given i.e. 81% utilization
Words with length 31 occupy 0,150KB of 0,204KB given i.e. 73% utilization
Total pseudo(including hash table) memory utilization: 92%
Total real(wordlist's words VS allocated block) memory utilization: 90/1000
Used value for third parameter in KB: 6300
Use next time as third parameter: 6292-
Time for making unsorted wordlist: 31 second(s)
Time for sorting unsorted wordlist: 16 second(s)
Ln 116, Col 57
INS
Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words - File Bro
File Edit View Go Bookmarks Tabs Help
Back Forward Up Stop Reload Home
Leprechaun_r13++_vs_Wikipedia_22,202,980_LATIN-Words
Name Size Type
Leprechaun.LOG 8.8 KB application lo
Leprechaun_r13++_generic_32bits.elf 628.7 KB executable
Leprechaun_vs_Wikipedia_LATIN-WORDS.lst 216 bytes plain text doc
Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.x 265.1 MB plain text doc
Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd.y 265.1 MB plain text doc
wikipedia-de-html.tar.wrd 93.7 MB plain text doc
wikipedia-en-html.tar.wrd 140.2 MB plain text doc
wikipedia-es-html.tar.wrd 30.4 MB plain text doc
wikipedia-fr-html.tar.wrd 36.0 MB plain text doc
wikipedia-it-html.tar.wrd 31.4 MB plain text doc
wikipedia-nl-html.tar.wrd 32.7 MB plain text doc
wikipedia-pt-html.tar.wrd 22.7 MB plain text doc
wikipedia-ro-html.tar.wrd 9.6 MB plain text doc
"Leprechaun.LOG" selected (8.8 KB)
[kaze@saturn:~/Docu...
Leprechaun_r13++_vs...
Leprechaun.LOG (~/D...


Visual C++ Toolkit 2003 Command Prompt
Size of Input TEXTual file: 23,830,432
\; Word count: 34,310,324 of them 21,978,966 distinct; Done
Size of Input TEXTual file: 10,073,451
/; Word count: 35,271,297 of them 22,202,980 distinct; Done
Flushing unsorted words ...
Time for making unsorted wordlist: 45 second(s)
Deallocated memory in MB: 1950
Allocated memory for words in MB: 266
Allocated memory for pointers-to-words in MB: 85
Sorting(with 'MultiKeyQuickSortX26Sort' by J. Bentley and R
Sort pass 26/26 ...
Flushing sorted words ...
Time for sorting unsorted wordlist: 17 second(s)
Leprechaun: Done.
D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13+-
dia_22,202,980_LATIN-Words>Leprechaun_r13++_32bits.exe Lepr
ATIN-WORDS.lst Leprechaun_vs_Wikipedia_LATIN-WORDS.wrd 5000
Leprechaun(Fast Greedy Word-Ripper), revision 13++, written
Leprechaun: 'Oh, well, didn't you hear? Bigger is good, but
Kaze: Let's see what a 4-way hash + 6,602,752 Binary-Search-
also the performance of a 4-way hash + 6,602,752 B-Tr
Size of input file with files for Leprechauning: 216
Allocated memory in MB: 1950
Size of Input TEXTual file: 98,215,517
¦; Word count: 7,375,373 of them 7,375,373 distinct; Done:
Size of Input TEXTual file: 146,973,879
/; Word count: 19,937,247 of them 17,322,675 distinct; Done
Size of Input TEXTual file: 31,913,244
¦; Word count: 22,829,701 of them 18,291,299 distinct; Done
Size of Input TEXTual file: 37,784,445
/; Word count: 26,296,387 of them 19,346,269 distinct; Done
Size of Input TEXTual file: 32,880,630
¦; Word count: 29,256,183 of them 20,331,005 distinct; Done
Size of Input TEXTual file: 34,311,298
¦; Word count: 32,128,367 of them 21,393,001 distinct; Done
Size of Input TEXTual file: 23,830,432
\; Word count: 34,310,324 of them 21,978,966 distinct; Done
Size of Input TEXTual file: 10,073,451
/; Word count: 35,271,297 of them 22,202,980 distinct; Done
Flushing unsorted words ...
Time for making unsorted wordlist: 45 second(s)
Deallocated memory in MB: 1950
Allocated memory for words in MB: 266
Allocated memory for pointers-to-words in MB: 85
Sorting(with 'MultiKeyQuickSortX26Sort' by J. Bentley and R
Sort pass 26/26 ...
Flushing sorted words ...
Time for sorting unsorted wordlist: 17 second(s)
Leprechaun: Done.
D:\Leprechaun_r13++\Visual C++ Toolkit 2003\Leprechaun_r13+-
dia_22,202,980_LATIN-Words>
Leprechaun.LOG - Notepad
File Edit Format View Help
[eflagrum[
]edopisni[
[ederekto]
[edelsens[
]eckzahns[
[eccarton]
[ecarboot[
]ebugtraq[ROOT
]ebeetley]
[eahivyle]
Above Binary-Search-Tree with MaxPEAK = 38 has NODEs = 58 and LEAFs = 15
Legend:
At left side of the word - '[' means no left successor
At left side of the word - ']' means left successor exists
At right side of the word - ']' means no right successor
At right side of the word - '[' means right successor exists
Bytes per second performance: 9,244,064B/s
Words per second performance: 783,806W/s
Input File with a list of TEXTual Files: Leprechaun_vs_Wikipedia_LATIN-WORDS.lst
Size of all TEXTual Files: 415,982,896
Word count: 35,271,297 of them 22,202,980 distinct
Number of Files: 8
Number of Lines: 35271297
Allocated memory in MB: 1950
Number of Trees(GREATER THE BETTER): 3410463
Forest population(Hash Function Quality regarding Collisions i.e. Hash Table Utiliza
Number of Hash Collisions(Distinct WORDs - Number of Trees): 18792517
Maximum Attempts to Find/Put a WORD into a Binary-Search-Tree: '38'
Total Attempts to Find/Put WORDs into Binary-Search-Trees: 121,674,042
Total Number of LEAFs in Binary-Search-Trees(GREATER THE BETTER): 7,990,635
Perfectly-Balanced-Binary-Search-Tree for MaxNODEs = 94 must have PEAK = 7 = roundir
Binary-Search-Tree(1st out of 1) with MaxNODEs = 94 has PEAK = 20 and LEAFs = 29
Binary-Search-Tree(1st out of 2) with MaxPEAK = '38' has NODEs = 58 and LEAFs = 15
Binary-Search-Tree(1st out of 2) with MaxLEAFs = 30 has NODEs = 93 and PEAK = 23
Words with length 01 occupy 0,033KB of 0,162KB given i.e. 19% utilization
Words with length 02 occupy 0,033KB of 0,162KB given i.e. 19% utilization
Words with length 03 occupy 0,040KB of 0,162KB given i.e. 24% utilization
Words with length 04 occupy 0,224KB of 0,646KB given i.e. 34% utilization
Words with length 05 occupy 1,311KB of 1,775KB given i.e. 73% utilization
Words with length 06 occupy 2,902KB of 3,549KB given i.e. 81% utilization
Words with length 07 occupy 5,345KB of 5,968KB given i.e. 89% utilization
Words with length 08 occupy 6,826KB of 7,581KB given i.e. 90% utilization
Words with length 09 occupy 7,683KB of 8,549KB given i.e. 89% utilization
Words with length 10 occupy 7,193KB of 8,065KB given i.e. 89% utilization
Words with length 11 occupy 6,606KB of 7,420KB given i.e. 89% utilization
Words with length 12 occupy 5,514KB of 6,130KB given i.e. 89% utilization
Words with length 13 occupy 4,599KB of 5,162KB given i.e. 89% utilization
Words with length 14 occupy 3,636KB of 4,033KB given i.e. 90% utilization
Words with length 15 occupy 2,900KB of 3,226KB given i.e. 89% utilization
Words with length 16 occupy 2,286KB of 2,904KB given i.e. 78% utilization
Words with length 17 occupy 1,763KB of 2,259KB given i.e. 78% utilization
Words with length 18 occupy 1,355KB of 1,613KB given i.e. 83% utilization
Words with length 19 occupy 1,065KB of 1,291KB given i.e. 82% utilization
Words with length 20 occupy 0,843KB of 1,130KB given i.e. 74% utilization
Words with length 21 occupy 0,659KB of 0,968KB given i.e. 68% utilization
Words with length 22 occupy 0,530KB of 0,807KB given i.e. 65% utilization
Words with length 23 occupy 0,418KB of 0,646KB given i.e. 64% utilization
Words with length 24 occupy 0,337KB of 0,484KB given i.e. 69% utilization
Words with length 25 occupy 0,278KB of 0,484KB given i.e. 57% utilization
Words with length 26 occupy 0,223KB of 0,323KB given i.e. 68% utilization
Words with length 27 occupy 0,182KB of 0,323KB given i.e. 56% utilization
Words with length 28 occupy 0,161KB of 0,323KB given i.e. 49% utilization
Words with length 29 occupy 0,131KB of 0,323KB given i.e. 40% utilization
Words with length 30 occupy 0,111KB of 0,162KB given i.e. 68% utilization
Words with length 31 occupy 0,100KB of 0,162KB given i.e. 61% utilization
Total pseudo(including hash table) memory utilization: 85%
Total real(wordlist's words vs allocated block) memory utilization: 114/1000
Used value for third parameter in KB: 5000
Use next time as third parameter: 4509-
Time for making unsorted wordlist: 45 second(s)
Time for sorting unsorted wordlist: 17 second(s)
Start
Visual C++ Toolkit 2003 C...
Leprechaun_r13++_(ELF...
Leprechaun.LOG - N...
EN
5:20 AM