A RAGMAN LURE, PRAM NAG AND LIDO

DOUGLAS C. GREENWOOD
Barrie, Ontario, Canada

While writing this article about computers vs. the English language, 1 searched my brain for some eye-catching title like King Kong Meets Godzilla. However, the confrontation is a somewhat more intellectual concept than such a title would connote. During the last ten years or so, there have appeared several articles about current attempts to manipulate English words by means of a digital computer.

To give the bottom line first: the computer loses. An obvious task that seems to beg for the ability of a computer is anagram solving. This is a relatively simple job nowadays if a computer is programmed to recognise words. A dictionary is analysed by the computer, creating a "signature" for each word. For example, the signature ACERS is assigned to acres, cares, races and scare; ACERT for cater, crate, react, and trace; and so on. This gives the computer an "Aha!" capability, which is to say, it has a wisp of intelligence because it can begin to recognise real words.

Now it can churn out anagrams. Unfortunately, we have an embarrassment of riches. Well, not all riches – actually, hundreds and thousands (millions for some multiple-word anagrams) of anagrams that are not worth reading. That is the trouble. For the computer to find an anagram that is really appropriate, in the case of even a short sentence, could take hours...years? Keefer (1986) notes that over three thousand anagrams were found by a seventeenth-century hermit for **Ave Maria, gratia plena,** Dominus **tecum** (Hail Mary, full of grace, the Lord is with you); but the computer would produce billions of anagrams from such a long sentence. Who has the time or inclination to search through such stuff as oh, **howdy, agriculturally isothermal fife,** as Morton (1987) sardonically asks? It all boils down to whether the anagram we create is the sort that, so to speak, takes one's breath away. There are anagrams and there are anagrams. The word stop, for example, yields **tops, post, spot, pots** and **opts,** but who needs them? How many inches of computer printout must one wade through to find such nineteenth-century pre-computer gems as **dirty room** for **dormitory, golden land** for **Old England,** or **best in prayer** for **Presbyterian?** Weed (1984) asks "Do we create [anagrams] in the hope that no other person – or machine – can match our performance, or do we create simply for the joy of personal discovery?" I'm sure the latter is the case for **Word Ways** readers – and certainly for crossword-puzzle addicts. He also points out that Louis X111 was an anagram nut who even appointed a royal anagrammist to

his court. Some have it, and perhaps they are right, that finding anagrams can be considered an **ars magna** - even for a **ragman**!

As a second example of computers vs. the English language, consider the pangram: a sentence using every letter of the alphabet at least once. The pangram is much harder to construct than the anagram; perhaps its anagram, **pram nag**, is more appropriate than at first sight. A well-known example: The quick brown fox jumps over the lazy dog. Another: Why jog exquisite bulk, fond crazy vamp? However,   more interesting than the pangram is the self-referential pangram, a concept so far advanced that one can accept the computer's help without demur. A self-referential pangram is one that not only contains every letter in the alphabet but states how many of them there are. Consider:

> This first pangram has five a's, one b, one c, two d's, twenty-nine e's, six f's, four g's, eight h's, twelve i's, one j, one k, three l's, two m's, nineteen n's, twelve o's, two p's, one q, eight r's, twenty-six s's, twenty t's, three u's, five v's, nine w's, three x's, four y's and one z

It is, of course, all very well to be wise after the event. But just try to compose such a pangram for yourself! For example, try it for a pangram which starts: This pangram includes... Although it can be done without a computer, this is where a computer should be enlisted to defeat King Kong. As reported by Weed (1984), Lee Sallows was challenged to produce an English translation of a Dutch self-referential pangram beginning Dit pangram bevat... ("This pangram contains..."), which is quite (if subtly) different from the one above.

Sallows designed a special-purpose computer to search for self-referential pangrams only, and set it working on October 3. Operating night and day, it was not until November 19 that the "Eureka!" light (as Sallows called it) went on. It had produced its first self-referential pangram:

> This pangram contains four a's, one b, two c's, one d, thirty e's, six f's, five g's, seven h's, eleven i's, one j, one k, two l's, two m's, eighteen n's, fifteen o's, two p's, one q, five r's, twenty-seven s's, eighteen t's, two u's, seven v's, eight w's, two x's, three y's & one z

The ampersand seems to have been a little devil in the woodpile, which the computer had accepted but perhaps a human would not.

As a third example of computers vs. the English language, I present LIDO, my acronym for "Literature In, Drivel Out". In a progress report "on the fine art of turning literature into drivel," Hayes (1983) set the computer to creating random generation prose from the letters of the alphabet, after the fashion of Sir Arthur Eddington's famous army of monkeys with typewriters.

I won't go into all the technical stuff, but what the computer had to do was (with heavy human help in programming, which a-mounted to just about the most blatant college crib ever) to create

prose after the fashion of various writers after excerpts from them were fed into the computer. When part of Hamlet was fed in, this emerged:

> How now, Ophelia? Nymph, in thy orison be all my heart; and it down he shall relish of it; then trip him, and the murder!

There is a lot more, but we must forgo its pleasure to report what I think is the choicest specimen of this high-tech garbage-out. When a passage from Faulkner's story **Two Soldiers** was fed in, the computer came out with:

> "Who let you in here?" he said. "Go on, beat it." "Durn that," I said. "They got to have wood and water. 1 can chop and tote it. Come on," I said. "Where's Pete?" And he looked jest like Pete first soldier hollered. When he got on the table, he come in. He never comes out of my own pocket as a measure of protecting the company against riot and bloodshed. And when he said "You tell me a bus ticket, let along write out no case histories. Then the law come back with a knife!"

And go tell Aunt Jody a bus ticket, too!

Molly Bloom's soliloquy from Joyce's Ulysses was also fed in. Out came:

> May thot to their yours chim jose ey eilly jused and hid yel the mark wask trooften herey ling sh thavered her inced I mea but day wom the eakin wips...

Wom the eakin wips? Well, it does sound pretty good to me – just like James Joyce. But then I'm afraid I'm a Joyce snob; I never could make out what it was all about.

So sayeth the computer. Better than an army of monkeys, I suppose, but I think us human writers is safe from computers – even if we don't write proper at least we will be understood.

Hayes, Brian. Computer Recreations: A progress report on the fine art of turning literature into drivel, Scientific American Nov 1983

Keefer, Bob. Anagram Solving in Pascal, BYTE Jul 1986

Morton, Mike. Recursion Plus Data Structures Equals Anagrams, BYTE Nov 1987

"Yank D. Weed" (anagram of A.K. Dewdney). Computer Recreations: A computational garden sprouting anagrams, pangrams, and a few weeds, Scientific American Oct 1984