

*Engineering*

Engineering

## General Observations

Engineers are the rubes of science. Unlike the typical physicist, who often has an interest in a wide range of subjects, including non-scientific subjects, e.g., history, and the literature and art of his time, the typical engineer is not only ignorant of such matters but he is proud of his ignorance. To an engineer, music is simply a means for enjoying stereo.

No truly educated person who has worked with engineers can in good conscience call an engineer “educated”. An engineer has been *trained*, albeit at a high level, by ignoramuses like himself. About psychology — meaning, here, the ability to recognize the feelings that produce various types of behavior in himself and others — he hasn’t a clue, even though this ability is of fundamental importance in the management positions he expects to be rewarded with.

It is probably impossible to explain to the layman how badly written, how *ugly*, most, if not all, industrial computer programs are. This is a direct result of the complete lack of esthetic sense on the part of programmers (i.e., software engineers), which in turn is a direct result of their lack of culture, which here of course includes lack of appreciation of mathematical beauty. Three of the ugliest cities in the U.S. — Troy, N.Y., Bethlehem, Penna., and San Jose, Calif. — are centers of engineering training, and, in the case of the latter two, of engineering employment.

“Ken:... One day you’ll realize there’s more to life than culture...There’s dirt, and smoke...”  
— *The Complete Monty Python’s Flying Circus: All the Words*, Vol. 1, Pantheon Books, N.Y., 1989, p. 23.

“The wrong of unshapely things is a wrong too great to be told;”

— Yeats, W. B., “The Lover Tells of the Rose in his Heart”

One proof of the fundamental narrowness, smallness, of the engineering mind, at least as it exists in the computer industry, is the fact that it did not recognize long ago the importance of “throughput”, i.e., the speed at which an entire job can be performed on a given computer — e.g., a job such as entering commands to instruct the computer to make calculations based on given data and then produce a printed report. (Thus, the time required for the typical user to find out what commands to enter, is a limitation on throughput.) Engineers love to fuss with details, and would much rather spend months increasing the speed of a circuit by a few billionths of a second without having any idea of how much this improvement will affect throughput, than they would in making a study to determine where, in fact, most of the time is spent in the various jobs that are performed using the computer.

The typical young engineer feels — with justification — that his management will be impressed by the degree to which he attempts to re-design, from scratch, everything needed for a new product, because that is the mark of a *hard worker*, an ambitious Team Player, a *creative* engineer. (Always *be busy!*) Yet the most intelligent and creative engineers — until they have been sufficiently discouraged — always try to do just the opposite, i.e., they always try to do *as*

*little* as possible in the way of new design to create a new product, preferring to rely as much as possible on off-the-shelf components whose reliability has already been proven.

“A good scientist is a person with original ideas. A good engineer is a person who makes a design that works with as few original ideas as possible.” — Dyson, Freeman, *Disturbing the Universe*, Harper Colophon Books, N.Y., 1979, p. 114.

A perfect example of the engineering mentality at work is the two-wheeled motorized scooter known as The Segway Personal Transporter. True, no one can doubt the ingenuity of the software and attendant hardware that keeps the vehicle balanced on its two wheels with a person aboard, but a perfectly good alternative, and one requiring no software or hardware ingenuity, would be a similar-looking device with a small third wheel behind for balance.

I have yet to meet an engineer who does not regard the task of learning to write well as a complete waste of time. Yet, even though many tasks that only a few years ago were staples of an engineering education, have been taken over by the computer — e.g., engineering drawing, doing calculations and certain low-level forms of problem-solving — the computer has achieved nothing remotely resembling competence at writing prose — not even the simplest technical prose. And here, one of the Dreyfus brothers’ arguments against artificial intelligence — that it cannot be achieved by machines because they have no knowledge or experience of what it means to “live in the world” — here, this argument may be applicable. What makes writing, in particular, non-technical writing, hard for people and impossible for machines at present is the fact that each word and phrase carries with it an enormous range of potential associations derived from the writer’s, and, hopefully, the reader’s, experiences in the world. Not only are many of these associations extremely subtle, but which ones dominate in a given passage usually depends on the context. I cannot imagine what the tabulation of these associations on a word-by-word, phrase-by-phrase basis would look like, or even if such a tabulation would be possible. What I am certain of is that the task of becoming a good writer is not one wit less difficult than the task of becoming a good engineer.

The habit of asking oneself, “Could a computer be programmed to do this?”, brings many surprises. Computers can perform most of the tasks required to assemble a car, but I doubt if any computer at present could be programmed to repair shoes.

## **Additional Thoughts**

Hell is being forced to spend your life working for your inferiors.

Among the many reasons I hope this age will one day be regarded with scorn, one is that it considered the computer hacker to be a model of intellectual brilliance.

The endless proliferation of high-tech junk in our time produces a kind of perpetual noise in the minds of those who design and use it. The worst thing that ever happened to computer science was the computer.

## Engineering

“The ethic here [in Silicon Valley] is too monolithic and intense about engineering and money. There aren't losers in Palo Alto. Some of the people I hang around with — like violinists from the Boston Symphony and neighborhood chess players — would be considered losers here.” — Peter Sprague, chairman of National Semiconductor, quoted in the *San Jose Mercury News*, Dec. 6, 1982.

First rule of mechanical engineering: every part is a moving part — it moves in the course of assembly, it may move during the operation of the machine it is part of, it moves when it is removed for repair, and it moves when it is discarded.

“A sledge hammer has only one moving part: it.” — Steve Martin

If you want to figure out how to do something, imagine you had to do the thing not once but many times.

In most jobs, standing still is a step forward. (And a step backward is merely standing still.)

Some people have a *physical* need to work. Even when virtually every moment of doing that work is anguish because they know that what they produce will be worthless, attempting to do nothing is, for them, even more difficult, more painful. If we can talk about people who are chronically sick, so can we talk about people who are chronically healthy.

Your work is your message to the world.