Computers For Smart People

Robert S. Swiatek

First edition

If you use material found in this book without permission from the author or publisher, we will send viruses and cookies – not chocolate chips, either – and spyware to your computer. We won’t burn down your village, but we will shut off your power food supply and spam you. Information of a general nature requires no action. When in doubt, contact the author. Mentioning him and the book is appreciated.

ISBN: 0-9817843-9-9

available only as an ebook

also by Robert S. Swiatek

Don’t Bet On It

***Tick Tock, Don’t Stop – A
Manual For Workaholics***

for seeing eye dogs only

***This Page Intentionally Left Blank –
Just Like The Paychecks Of The Workers***

I Don’t Want To Be A Pirate – Writer, maybe

wake up – it’s time for your sleeping pill

***Take Back The Earth – The Dumb,
Greedy Incompetents Have Trashed It***

Press 1 For Pig Latin

***This War Won’t Cost Much –
I’m Already Against The Next One***

here's your free gift – send $10 for shipping

Mirror, Mirror, On My Car

Save The Animals And Children

Recipes For Joy In Life

I’d like to thank all the people who made this book possible, in particular, all those people I met during my stay in Binghamton in the early 1970s. I especially thank my niece, Elizabeth Thomann-Stellrecht, who was responsible for the great cover of this book. Her work can be found on the vast majority of the books that I published since the spring of 2008 – that’s much more than a two-thirds majority needed to overcome Republican objections. Over the last few years, people have raved about these covers at various arts and crafts festivals of which I have been a part. Some have even purchased a book. Thanks!

To Pat and Lou

Table of contents

Introduction

I began writing my very first book in September 1972. It dealt with computer concepts and was meant as a high school math textbook to teach programming. It used APL, which stands for *A Programming Language*, a highly scientific language for the computer. At the time, a few publishing companies expressed interest in the book but as the days passed, they declined in getting the book on the market. I wasn't completely discouraged.

Their excuse was that there were enough of these types of books out there and I accepted that. At the same time I saw a dilemma insofar as books written about fairly common subject matter would not get printed for this same reason but revolutionary topics probably wouldn't make it to print either because the publisher wouldn't want to risk getting into an untested, unknown area. I never did submit it to a far-out press, even though this was just after Woodstock.

I did use the book when I taught a programming course in high school shortly thereafter, in addition to the regular APL textbook. However, once I left teaching the book was stored away gathering dust, rarely to be perused. Over time I realized that there was no chance that it would ever get published in its existing form. I also thought that it could be revised, with the original language of APL replaced by a common, understandable language. In this way it could have relevance. Of course, that meant almost a complete rewrite of the book.

In August 2001 on a Sunday afternoon I decided to dig out the book and redo it. I went through it but decided not to do it. The next day I changed my mind. I wound up revitalizing and resuscitating it using a generic language. This turned out to be a language that I created, utilizing features of many computer languages that I had come in contact with over the years. Since all languages do basically the same thing but by different means, I took all the benefits of each language and combined them into my language. The book would now be used to illustrate what computer programming is all about to people unfamiliar with the subject.

The intent of this book is to reach two types of people. The first are those who would like to get an idea of what programming is all about since that may be what they want to do as a profession. The other person to be reached is that individual who has little computer knowledge but would like some insight into what programming involves. This breakdown includes a great number of people.

By no means is this book meant to be a computer text but rather a means of spreading knowledge about computer programming. My goal is to make clear each topic presented but anyone reading the work need not feel disappointed if some area of the text is not completely comprehended. However, by the time someone is through with the book, it is my goal that either they will decide to pursue this field or at least have some basic understanding of what programming is all about.

Despite the possibility of getting this book published so many years ago when I first wrote it, there were a few things missing. As I mentioned, the language was too difficult for most readers to comprehend. Also, since it was my first book, it was missing what could be found in the books I wrote after it, namely at least a small amount of humor. The subject matter may have limited that aspect, but as I have found, there are very few books where levity can't be interjected.

In general, it is probably better that the book didn't get published at that time. It really wasn't ready to come into print. However, when I revised it in 2001, all these limitations would be gone. Half a dozen years later, the work still wasn't published. I did some more modifications in January 2010 while staying in my cousin Jim's town home in Sun City Center, Florida. Incidentally, I have at least three cousins with that name. Then in December 2011, I decided to publish it as an ebook, resulting in a great deal more editing. Since I had created my own computer language, that created the biggest holdup. I felt for the longest time that the programs – few though they were – had to be thoroughly checked over since they couldn't really be tested with a computer. I needed to put in the effort to get this task done.

Somehow, I came up with a new idea. Every program found in this book is here for instructional purposes. It is meant to display computer concepts and who really cares if there are a few places where something may fail or could be done better. In reality, these programs have nowhere near the bugs that you will find on the Internet or even on your PC, each of which is rushed into production without checking. As you can tell, that approach was not done in this book. After all, quite a few years had passed since I started writing it. For that reason, any problems that you encounter in this work are truly minor and can easily be overlooked. If you are a person who likes to correct others by finding tiny mistakes in books, I need only remind you that every book that was ever written has at least one error, of some kind or another. Don't waste your time.

As far as the book title that I first chose, this goes back to my days at Binghamton University when I was studying for a degree in computer science. My fellow students and I worked together as a team to get projects done. The effort required was intense but we had a good sense of humor about it. In fact while going through the degree program one of my study-partners remarked, "Six months ago I could not spell computer programmer – now I are one!"

We all got a laugh out of that, and I loved that title. However, I decided that there wasn't enough room on the cover to put all those words – if I used a smaller font, not many people could read it – so I thought about another one that would be better. This didn't come easy, but eventually I settled on ***Computer For Smart People***. I hope you find this treatise to be enjoyable and enlightening.

1. Elements of language

Any language that we come in contact with follows certain rules. This applies to Spanish, English or any computer language. Naturally the fewer rules there are, the easier the language. As the number of rules increase, so does the difficulty. Unfortunately there may be no choice but to have a preponderance of rules, such as the language of a computer system. However, I shall get into that later.

For now, let us talk about the language of English, although you will soon realize that what applies here will be the same for any language we consider. We have to start with certain basic symbols, specifically the letters of the alphabet that apply to this language. In our case they're the letters *a* through *z*. But we also need to mention the capital letters, *A* through *Z* as well as certain punctuation, such as the comma, period, question mark and a few other symbols. I think you get the idea. Our character set will be around 75 different symbols. As we progress we shall be introduced to more and more of them.

These elements or basic symbols will be put together to form words. Thus the letters "t", "h" and "e" form the word, "the." Some letters put together may not form a valid word, such as "q", "j", "x" and "h," no matter what order we put them in. You might reply that you went to school with a guy from Russia whose name was exactly those letters in that same order, but that doesn't count. Some combinations will give us words while others may not. There could come a day when the four letters we mentioned form a valid word, since new words come into existence from time to time in the English language.

A few examples of words that feature symbols other than our usual letters of the alphabet are "son-in-law" and "o'clock," and just recently one of my favorites, "24/7." Thus we need the hyphen, apostrophe and the slash, since some words use these characters. We will run into situations where other symbols will be used, which will be discussed when needed.

You might ask how it is determined whether a combination of letters is a valid word. This is decided by predefined rules of the language. By referring to an English dictionary, you can see whether you have a word. If you care to read an appropriate book, check out ***The Professor and the Madman*** by Simon Winchester. It's a tale of murder, insanity as well as the making of the Oxford Dictionary.

As you can imagine, there are various editions of the dictionary as well as those produced by different companies. This will mean that one dictionary might indicate that one combination of letters is a word while another may not have it listed. This difference along with the additions and deletions of words to the dictionary with each passing day adds to the complexity of the language, which we will not be burdened with.

To temporarily bypass this dilemma, we shall use one dictionary only and thus there will be a final say on whether or not a certain combination of symbols is a word. When taking a group of words together, we next form what is referred to as a sentence. Not all combinations of letters form valid words, and similarly not all combinations of words form valid sentences. Again the determination is based on certain rules, which can be found in various books on the subject.

As you can tell there are many rules. That may be why English is such a difficult language. The rules don't end here, as now sentences will be put together to form paragraphs. Not all combinations of sentences will form meaningful or valid paragraphs and once more we need to follow guidelines, which have been set up in defining the language. But assuming we have some valid paragraphs, these put together will make up what is referred to as a chapter. Obviously there are more rules in determining this composition, just as before.

Now taking a group of related and meaningful chapters, the combination will result in a novel or work of nonfiction. We now have what is referred to as a book and I shouldn't have to remind you of the necessity of following certain rules in order to achieve a meaningful book. The last grouping will give us our library, that is, putting a set of books together gives us this structure. Assuming all our books pass the test of "validity," at this point we have no special rules as to what can go into our library.

Some might say that I missed a few groupings such as putting words together to form a phrase. What about bunching three novels together for a trilogy or a set of works together to get a volume? Why not put all the psychology books in one department and young adult fiction in another? You would have a very valid point but I am just trying to outline the main tenets of a language. As I said earlier, all languages will follow a similar set of rules, whether they are a foreign language or a computer language.

Just because there are rules for forming valid words and sentences and the like doesn't mean that everyone conforms to them. I have worked with many people who make up words. I'm sure you have too. These individuals use so-called "words" and "sentences" as though they were as common as all those that are valid. This does make for frustration and confusion and lack of understanding. However, it does gives me plenty of material for my books. If you've read any of them, you're aware that I have a great deal of fun making up words. Someone has to do it. My 2005 book, ***for seeing eye dogs only*** and its two sequels deal with missing intelligence as well as oxymorons, acronyms, pleonasms, words and near words. There's another combination that I just heard about recently, but it's not included here because I can't spell it.

Corporate America has its own set of words and phrases, but good luck finding documentation anywhere. This makes it extremely difficult to figure out exactly what they mean. If you are part of the business world as I had been for over twenty-five years, mostly as a consultant, you may find it difficult in determining what people are talking about. If you are outside the environment and you try to understand what is being said, it's even worse. Perhaps that's why big business has so many problems.

If a language has no rules, you will never be able to use it or understand it. Too many rules mean that there will be rebellion and once again you may as well have no rules. Obviously there have to be some rules but there is a limit. You can't have too few precepts nor too many. That middle ground will result in a successful language that meets the needs of a group of people. This applies to everyday communication as well as the languages of computers.