

My Life as a Forth Interpreter

By K.

Languages come easily to me. My family is from the very middle of *Mitteleuropa*, where countries are small and going out to buy a loaf of bread can take you across a linguistic frontier. At home we spoke German and Czech interchangeably, and when Father wanted to keep something from the children he would resort to French—which gave me a powerful incentive to learn that language too. My Uncle Gregor, who was a commercial traveler for some years (until an unfortunate accident confined him to his room), talked with shopkeepers in seven languages, including Turkish. I myself, though no scholar, mastered the classical languages and Russian before I left the gymnasium, and in the years since, of course, I've added English.

It was only natural, then, when I found

myself in a strange land and in need of a livelihood, that I turned to translation. At first I worked for a New York bank, handling correspondence with the Frankfurt office. It was a dull job, and neither well paid nor well respected. (The boss was continually saying *Bitte* and *Danke* to me, which I found insufferable; he knew not another word of German.) What I really wanted was to do simultaneous translation, as I had heard it done on lunchtime visits to the United Nations.

About a year ago the international department of the bank was reorganized, and I was left without a job. I made inquiries at the U.N., but I lacked credentials. And so it happened, one spring day, that I answered the following newspaper ad:

INTERPRETER

Fast, experienced. High-tech assignment. Enter the exciting world of computers. 32K.

The woman at the employment agency gave my resumé a cursory look and asked a few questions. Curiously, we talked mainly about Polish and about something she called Reverse Polish, which I gathered was a sort of backwards jargon, like pig Latin. I told her plainly that I could speak no Polish either backwards or forwards, but it made no difference; all of Central Europe was of a piece to her. At the end of our brief interview she said I could have the job, and she scribbled the address on a slip of paper. It was a remarkable address, being made up of nothing but numerals. What the numbers meant, I eventually decided, was 1100 110th Street, 11101-1011.

Tracking down my new employer took me into an area of the city I had never visited before. The cab driver, a Ukrainian emigré, assured me he could find the place: it was out on the island, he said, beyond the airports, in one of the new industrial "parks." And indeed as we drove on through the twilight we entered what looked to be a factory district. The glass spires of Manhattan had been left behind; here the buildings were low, flat-

Illustration: Bud Thon

roofed, windowless and uniformly rectilinear. Yet they were not entirely without ornament. On the two long faces of each building was a colonnade made up of a dozen or more pillars set close to the wall.

In stilted English my driver told me the story of his life. (I concealed my knowledge of Russian lest I hear it in even greater detail.) At one point he interrupted himself to say, "It is old joke about America: The streets are paved with gold." When I looked out the window, I saw what he meant. We were on a parkway, which wound among the factory buildings and occasionally even passed under them. I counted 16 parallel lanes, separated from one another by narrow grass median strips. And yes, the roadways did have a metallic gleam, although the color was more coppery than golden.

The building where the taxi left me was one of the larger examples of the new industrial architecture, with a 20-column colonnade front and back. Inside it seemed bigger still. The entire space was filled with brightly painted machinery. One section had thousands of small machines, all alike—they looked like parking meters or gumball dispensers—laid out in columns and rows as in a perfectly planted orchard. Elsewhere the apparatus was arranged in banks that appeared to work in unison; I supposed they must be stamping out multiple copies of the same part. Near the middle of the plant I could see a control panel with hundreds of dials, levers, switches and colored lights. And toward the four walls were a number of cleared areas that looked like helicopter landing pads; I took them to be loading docks.

In truth the place was not unpleasant in the ways I would have expected of a factory. There was no noise, no clangor of metal meeting metal, but only a continual whoosh in the background, like the sound of a radio tuned between stations. And there was no soot or grime; on the contrary, the room was exceptionally clean and bright—even cheering in a gaudy

way. But why had they sent for me? What need could they have here for a translator?

A guard examined the address written on my slip of paper and conducted me to a glass-walled office overlooking the factory floor. The legend in gold leaf on the door panel read "Operating System Supervisor." Inside, a woman in a tweed suit gave me a brisk handshake. "You're the new interpreter?" she said. "I'm Ms. Dos. I wish we had time to get properly acquainted, but I'm afraid we're going to have to put you to work straightaway. He's waiting."

I followed her down an open stairway onto a metal catwalk suspended 20 feet above the factory floor. It was narrow enough that we had to go single file, and so she spoke to me by half turning around as she walked. "It's a good thing you showed up when you did," she said. "I don't know what I would have done. We just lost our last interpreter. Clobbered by a bug." She must have noticed a flicker of alarm in my face. "I know it's unsettling to be brought in under these circumstances. But, as you doubtless know better than I, it's an occupational hazard."

We were beyond the gumball orchard and somewhere to the right of the control panel when Ms. Dos took a sudden turn into an unmarked byway and then descended a ladder to the polished floor below. I followed, gingerly. "Here we are," she said, with a sweep of the arm.

"Now I realize you're fluent in several languages, but I don't think you've ever worked with a threaded one before, have you?"

"Pardon me?"

"Never mind. I'll run through the whole routine anyway, just to be on the safe side. There's not much to it, really; you just have to keep track of a couple levels of indirection, and of course be sure you don't forget anything. The Outer Interpreter, you know, is the heart of this operation. We're all depending on you."

"OK, so the tokens come down this chute here, one at a time. The moment a token arrives, you take it, you read the word on it, and then you look up the word in the dictionary. We keep the dictionaries over here." She riffled the pages of a great folio volume mounted on a lectern.

"Sometimes you may have to search more than one dictionary, but I'll come back to that."

"Anyway, let's assume you find the word. Now there's quite a lot of information in a typical dictionary entry, but most of it need not concern you. And to make the job more manageable we've color-coded the dictionary entries. It's red for constants, yellow for variables and blue for colon definitions. So you take one of these slips in the color that matches the entry, and you read down through the definition until you come to what we call the PFA, the parameter-field address. You write the PFA on the slip, then you come over here and put it in a capsule of the same color. Finally you drop the capsule into the tube that matches its color. All you have to do is keep your colors straight."

"Where do the tubes go?"

"The red and the yellow ones go to the stack supervisor, who has his boys push a value onto the stack. The blue one is a direct line to Mr. Next. He takes care of everything else. Any other questions before we go on?"

"Surely," I said, "it's not necessary to look up every word in the dictionary." I was miffed that she would take me for that kind of translator.

"How else would you find the PFA?"

"Well, suppose I already know what the words mean."

She looked at me in sudden confusion. "What they mean? How could you possibly know that? I don't know what they mean. Even Mr. Next doesn't know. No one knows what the words mean—unless, I guess, He knows."

"He?"

"Well I always think of Him as He, although it could just as well be She. You know. Up there. Beyond the glass."

I nodded assent, the way one does with people who are obviously mad. Ms. Dos had evidently developed similar suspicions about me, and she continued her lecture in a tone appropriate to a child or an idiot. "Now there will be times when you don't find a word in the dictionary."

What you do then is you take the token itself and you put it in this slot, which goes downstairs to the numbers department. Ninety-nine percent of the time that's all there is to it; the counting house will take care of it. But, and this is very important, if you should get a token that isn't a dictionary entry and isn't a number, that means we have a bug. What you do then is pull the alarm and get out of here. Quick." She showed me the alarm, a red box near the ladder. Cast into the metal were the instructions: "Break Glass. Pull Lever."

"There's just one more thing you have to do after processing each token. Before the next token comes down the chute, you go around the corner here and you check the stack." She showed me. It was a stack of cafeteria trays, spring-loaded to keep the top tray level with the counter. There was no telling how deep it might be. "That stack should never be completely empty. There's a bottom tray on it like the last \$20 bill in the cash register, that you never take out unless you're being robbed. Anyway, if the stack is empty that means we've got a different kind of bug, and you pull the alarm box over here."

The talk of bugs unnerved me. I have a morbid fear of insects and spiders, and although I know it's irrational, that makes it no less real. I come by it honestly, anyway: my mother was far worse. When she found a cutworm on the lettuce or—God forbid!—a cockroach on the kitchen floor, she would freeze. She couldn't kill it; she couldn't run; she couldn't even scream. This phobia of hers had something to do with her brother—my Uncle Gregor—but I never learned exactly what. Whenever I asked her, she just shivered.

I could have walked out that first day. I could have said to Ms. Dos, "No thank you, this is not the job for me. I am a professional translator; I know a word from a number without looking in a dictionary." The main reason I stayed, thinking it



would be only for a day or two, is that I was intrigued. When I left home for Amerika (my other uncle, Franz, always spelled it that way), I was told I would see wonders. I was told they had a machine in Amerika for everything—one to wash the clothes, another to beat an egg, a third to brush your teeth. But no one at home could have guessed that I would find myself operating the machinery of a vast language factory, where words and numbers tumbled down chutes, got sucked into pneumatic tubes and were pushed into coin slots. If this is how an interpreter worked in the New World, I thought I had better find out about it.

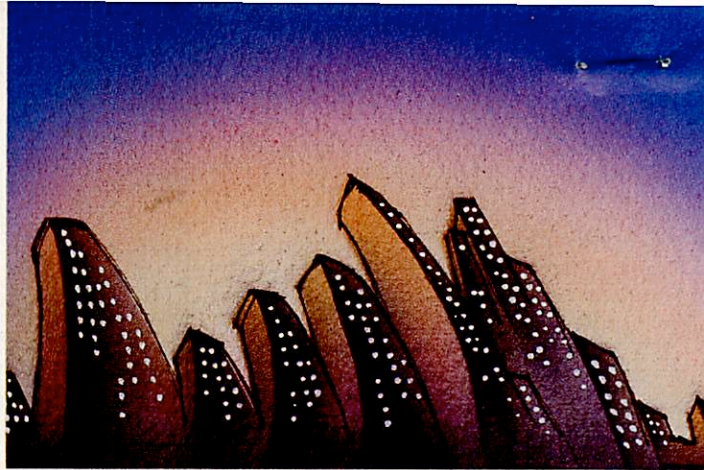
Moreover, the job gave me a chance to add another language to my repertory, and what an odd language it turned out to be. I couldn't identify it, but I knew it was not Polish. Many of the "words" inscribed on the tokens were not words at all but mere marks of punctuation—a period, a colon, an exclamation point, and so on. The dictionary was equally strange. It was not organized alphabetically. Instead, you looked up a word by starting at the back of the book and checking the last entry; if it didn't match the word you were looking for, it would tell you the page and line number of the next entry to be checked. You worked all the way through the dictionary by following this trail of cross-references.

The words themselves were baffling. At first I thought Ms. Dos must be right: they were completely senseless. One fragment I remember from those early days said, *FLUSH OVER OVER HERE*! When I

would see such demented ranting, I would get the giggles. Later, though, phrases began to tantalize me with glimmers of meaning. A series of tokens would come down the chute, saying, *DUP = IF ROT ELSE DROP THEN*. This was a dark saying but not totally opaque. Apparently someone was either going to rot or going to drop—but who? A dupe?

For some weeks my own tasks seemed as arbitrary and meaningless as the words I was looking up. The PFAs were mere numbers that had no obvious connection with the words they supposedly defined. My position near the stack and the dictionary, however, turned out to be a good observation post, and in time patterns began to emerge. I noticed that whenever I sent a message through the red or the yellow tube, the stack handlers would come and put an extra tray on the heap. Similarly, when I passed the PFA for *DUP* through the tube to the mysterious Mr. Next, the stack boys would make a copy of the top tray and add it to the stack. When I processed a *DROP* token, the top tray would be discarded.

Other words had effects of more direct consequence to me. When I came upon a colon in the stream of tokens, a clerk from the next room would shortly appear and begin adding definitions to the dictionary. The word *FORGET* brought the clerk back to rip out entire pages. (The first time this happened I tried to stop him—I was sure one of us would get in trouble—but he assured me it was all part of the job.) Fol-



lowing the word *DEFINITIONS*, a clerk would set up another lectern for a new dictionary.

It was not hard work. The tokens—bright brass coins that not only looked like subway tokens but in fact worked perfectly well in the turnstiles—arrived at a steady pace. I would go through my ritual of consulting the dictionary, shooting a message off to Mr. Next or whoever and checking the stack, and when I got back the next token would just then be rattling down the chute. There was an element of tedium to it all, particularly in the constant references to the dictionary, but it was the kind of work one can do with half a mind, while simultaneously planning dinner or composing a letter home.

The one relief from the routine was the bug alarm. The first time I had to pull the switch I was frightened, not least because I almost missed my cue. I had looked up a word and been unable to find it, which happened fairly often, and so I had dropped the token in the numbers slot. This time, however, I did not hear the usual clanks, clunks and ratcheting noises as the token worked its way through the machinery; instead, it merely reappeared in the reject bin below the slot. I very nearly did what one always does when a balky vending machine won't take your quarter: try it again. I had the token in my hand when I remembered. I looked around warily for signs of six-legged life, but the floor was spotless. All the same, I lunged for the alarm box and smashed the glass with the little red hammer that hung on a chain.

At the top of the ladder I found the catwalks filled with milling workers. A siren was whooping but no one had panicked or even seemed to be in much of a hurry. There was no question of shoving through the crowds on those narrow walkways, and so I shuffled along with the rest, toward one of the loading docks and then into the open air. After 10 minutes or so the all-clear sounded and we filed back in.

I learned soon enough that bug alarms were not the rare emergencies I had supposed they would be. We had at least two or three a day, and sometimes they came

in such quick succession we barely had time to get back to work before the next one sounded. Most of the employees took them rather like school fire drills—as an officially sanctioned occasion for goofing off.

It was during the bug drills that I first met some of my coworkers: the stack boys, who were a good-hearted rough-house gang, the clerks from the lexicography department, who were constantly in and out of my office revising the dictionaries, the text readers who worked directly over my head, and the accountants from the numbers department, one floor down.

In brief conversations outside the gates I picked up a good deal about the operation of the plant. The stack boys told me, with more relish than tact, about the fate of my predecessor. It seems the Outer Interpreter's routine checks on the stack can detect one possible malfunction, called a stack underflow. The opposite problem of stack overflow cannot be caught until it's too late. The last Outer Interpreter had been trapped in his room by a particularly sudden and massive overflow and crushed under tons of cafeteria trays. I resolved to keep a closer eye on the stack.

The countinghouse crew I could never warm up to. On their breaks they stood around playing some numerical parlor game. (It was something like hangman, but instead of guessing the letters of a word, you guessed the prime factors of a large number.) The text readers were more congenial company. It was they who supplied the tokens that tumbled down my chute. Their job was to scan an incoming stream of text and break it up into individual words, throwing out the intervening spaces and stamping each word onto a brass blank. I asked where the text came from in the first place. "From Him."

Such passing references led me to some tentative conclusions about Him. Evidently I had stumbled onto a small religious cult. Virtually everyone in the plant believed (and the text readers believed with much fervor) that somewhere above was a powerful unseen being who was the author of all words. He wrote them in green fire on the glass lid of the sky. Our lot, here below, was to do His

bidding not from fear of punishment or for hope of reward but simply because there was no alternative. I have little patience with a doctrine that leaves so little room for free will, but this religion at least had a measure of humor. The deity of the cult was omnipotent but not infallible; indeed He was presented as something of a chronic blunderer, always sending us on errands that were bound to come a cropper. As one of the readers put it, He is the source of all words *and* of all bugs.

I heard rather more about Him than I cared to know, but inquiries about the enigmatic Mr. Next got me nowhere. None of my colleagues admitted to knowing him. There was another mystery on my mind as well. If I was the Outer Interpreter, there simply had to be an Inner Interpreter. I knew I would not fully grasp this new language until I had found him.

I had been on the job six months when I began taking shortcuts. It was not laziness that led me to this; on the contrary, I felt I was making a natural and obvious improvement in my own efficiency. By then I had long since memorized the PFAs of all the commoner words. (How could I help learning them, when *DUP*, *SWAP*, *ROT*, *DROP* and the like came by me a hundred times a day?) It seemed idiotic to continue going through the rigmarole of a dictionary search for every word. The searches were made all the more onerous by the fact that most of the common little words were defined near the front of the book, so that I had to follow a long trail of word-to-word links to get to them.

As a first step I dispensed with following the chain of cross-references. I just flipped directly to the page where I knew the definition would be found. Soon I dropped even this pretense of following the rules. If a colon or an exclamation point came down the chute, I already knew the PFA, and so I just wrote it on a blue slip and shipped it off to Mr. Next. There were still unfamiliar words I had to

BD Software, Inc., maker of the original
CP/M-80 C Language Development
System, knows

Time is precious

look up—indeed, brand new ones were appearing all the time—but I was soon translating more than half of the tokens without cracking the books.

I started applying common sense to numbers too. They had been a major irritation. I would get a token that said "137," and although I knew perfectly well it was not in the dictionary, I had to check every word on the list before I could dump the token in the slot. On most days He sent down a great many numbers, and I grew thoroughly sick of these long and fruitless searches. Hence I began culling the numbers out by hand and passing them on directly. Some care was needed in this. For example, "+1" was a number meant for the coin slot, but "1+" was a "word" with a PFA that had to go to Mr. Next.

My colleagues were not happy with my new work habits. As far as I know, they never learned how I was doing it, but they couldn't help noticing a terrific increase in the pace. Most of my time had been spent leafing through the pages of that blasted book. With the bottleneck removed, the lexicographers, the stack boys and even the text readers were being deprived of their leisure. Over a cup of coffee it was hinted, with a broad, embarrassed smile, that I should take it easy. When I didn't comply, the suggestions became more emphatic. People stopped talking to me although not about me. The possibility of a union grievance was mentioned.

Apart from disgruntlement, however, my new methods had no ill effects. I was enjoying my work for the first time; what I was doing now was less mechanical and closer to the real art of translation. Besides, I was sure my improved performance would be noticed by those in charge, who might therefore view me as a candidate for promotion, ideally for a move "inward."

As the weeks passed I grew more confident. I began calling bug alerts on my own authority. A token would come down

So the compilation, linkage and execution speeds of BDS C are the fastest available, even (especially!) on floppy-based systems. Just ask any user! With 15,000+ packages sold since 1979, there are *lots* of users . . .

New! Ed Ream's RED text editor has been integrated into the package, making BDS C a truly complete, self-contained C development system.

Powerful original features: CDB symbolic source-level debugger, fully customizable library and run-time package (for convenient ROM-ing of code), XMODEM-compatible telecommunications package, and other sample applications.

National C User's Group provides direct access to the wealth of public-domain software written in BDS C, including text editors and formatters, BBS's, assemblers, C compilers, games and much more.

Complete package price: \$150.

All soft-sectored disk formats, plus Apple CP/M, available off-the-shelf. Shipping: free, by UPS, within USA for *prepaid* orders. Canada: \$5. Other: \$25. VISA, MC, COD, rush orders accepted.

BD Software, Inc.

BD Software, Inc.
P O Box 2368
Cambridge MA 02238
617 • 576 • 3828

Escape the Programmer's Prison

Do you feel imprisoned by your IBM PC? Do you spend too much time wrestling with a rigid, unyielding programming language? Are you tired of clumsy programming tools that straight-jacket your creativity instead of liberating it?

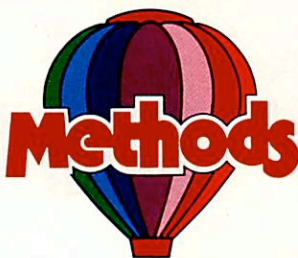
If so, you're ready for Methods—a Smalltalk, object-oriented programming system for the IBM-PC and compatibles that sets you free from the constraints of other languages. Until you program with Methods, you don't know just how quick and creative programming can be on the IBM PC.

Methods is...

... high-performance object-oriented programming. A powerful, language-compatible subset of Smalltalk-80™.

... a toolkit of over 100 classes—easily-customized software building-blocks for rapid, incremental development of real-world applications.

... an open-ended, window environment that encourages exploration while allowing recovery from any error.



... extensively documented in a four-part manual for everyone from beginners to experienced programmers—the perfect introduction to Smalltalk programming.

... ideal for software prototyping, simulation, databases, advanced user interfaces, and numerous AI applications.

Methods also offers easy access to DOS, a powerful directory/file browser system, remote UNIX™ access, an object-oriented shell for DOS programs, and much more.

Methods requires DOS and 512K RAM on IBM PC's (including AT) or "compatibles," and can be used with or without a mouse.

digitalk inc.

5200 West Century Boulevard
Los Angeles, CA 90045 (213) 645-1082

Available from Digitalk for \$250. Outside U.S. add \$15.00 for shipping and handling.
California residents add sales tax.
Educational and dealer discounts available.



IBM is a trademark of International Business Machines Corporation.
Smalltalk-80 is a trademark of Xerox Corporation.
UNIX is a trademark of Bell Laboratories.

CIRCLE 36 ON READER SERVICE CARD

with some bit of nonsense stamped on it, which I knew immediately was neither word nor number. Rather than waste my time with the dictionary or the numbers slot, I just broke the glass and headed for the door.

Finally I began making silent corrections. One day the word *DORP* came across my desk. There was no point in looking it up or trying to pass it off as a number, and I was tired of fire drills. I knew well enough what He meant, and so I did Him a favor. I put through the PFA for *DROP*.

I still don't know how I slipped up. For some days I had suspected them of testing me. The same sequence of words kept coming through over and over, sometimes with small variations. As soon as I caught on, I made a special effort to be consistent, to treat each word the same way every time, to make the same emendations, to pull the alarm in response to the same errors. It was like trying to sustain a complicated lie under prolonged interrogation. I may have gotten confused at a critical moment. It's also possible someone inside betrayed me.

I was alert to danger. I kept a wary eye on the stack and on the stack boys, who had become openly surly. But the ending they had planned for me was subtler than that.

Early in the day there was a long series of undisguised test words. I knew what He was up to. Several times He added a set of terms to the dictionary, then issued a *FORGET* to delete some of them. I had to keep track of which words were defined and which had been forgotten at any given instant.

About lunchtime there was a pause, then a few more words tumbled in, at a slower rate than usual. There was some business about moving a fence; when that had been accomplished, another *FORGET* arrived. The word that followed was one I had not seen before: *INTERPRET*. I held the token in my hand, I went to the dictionary, I followed the chain of references for page after page. Finally, about halfway to the front, I found it. Coming upon that word in the dictionary was like seeing my own name in the telephone book. There's something grand about such public testimony to your own existence, but it can also make you queasy. If you're in the book, anyone can get to you. And a word, a name, is so easy to blot out.

At that moment I knew I was finished. I was to be the instrument of my own execution. I was to write down my own PFA on a blue slip and stuff it down the blue tube. In a matter of seconds the lexicographers would appear and methodically rip me out of the dictionary.

The guards caught up with me in the open bay by one of the loading docks. They marched me off to another glass-walled room, where the sign on the door read: "Office of the Inner Interpreter." A blue tube entered through the back wall and descended in a smooth arc to a bin by the desk.

Mr. Next and I had a long talk. He told me about his own career with the company and tried to explain the duties of his office. From the description he gave me, I might have guessed he was in charge of the mailroom: it was all about current addresses, forwarding addresses and return addresses. He worked with the list of numbers that made up the bulk of each dictionary definition, the part I had had no cause to read. The numbers, he said, were addresses of still more words. His job had two parts. First he found the next address in the list, the one following the PFA we were working on currently. He saved this forwarding address and gave the current address to his executive assistant. When the assistant came back, Mr. Next would take out the saved forwarding address and treat it as a new current address. He would read one more number down the list, getting another forwarding address to save, pass the old saved address to the assistant, and so on. The process continued until he came to the end of a definition.

There was more. Mr. Next and his staff kept their own stack, similar to mine but holding only return addresses. Sometimes, he said, platters were transferred between his stack and mine, although I had never been aware of it. I was able to follow this description only in the foggiest way. Much of it failed to register simply because I didn't see the point of it all. The troubling thing is, I don't think Mr. Next did either. In some ways he had even less understanding than I did of the language we were supposed to be interpreting. He could not tell me the meaning of one word.

Eventually it came time to discuss my own troubled circumstances. Like figures of authority everywhere, from fathers to principals to presidents, Mr. Next was more hurt than angry, more disappointed than vengeful. "You showed great promise, great potential, K. We all had the highest expectations. From the moment you arrived you were quick, accurate, conscientious . . ." He made spider motions with his hands as he talked. "One

day you might very well have found yourself in this office, behind this desk."

I knew well enough what this tone implied. I wasn't sorry to be leaving. As a matter of fact, it seemed to me a great dispensation: I would never again have to *DUP*, *DROP* or *ROT*. I put up a fight only out of a sense of fairness. "I think I deserve some recognition for what I've accomplished here," I said. "The efficiency of this plant must be up tenfold. You might weigh that in the balance along with my mistakes, whatever they were. At the very least I deserve a good recommendation."

Mr. Next made the saddest eyes. "I wish I could do that for you, K. Or I wish

you could understand why I can't do it. It's not just a matter of mistakes; you did make a few, but we could forgive them and work with you to correct them. No, it's something more fundamental. The truth is, K.—there's no hiding it—you have become a bug." ■

"K." is a pseudonym of (among others) Brian Hayes, a Minneapolis writer who sometimes dreams of becoming an Inner Interpreter.

C Programmers: If you do business application development, you need

FAST PROGRAMMING™

The C programming language tool for business applications.

Why choose Fast Programming?

Because it is a **complete** development system. You will no longer need to integrate incompatible libraries and fragments of programs.

Because finished, ready-to-compile C programs are generated.

Because there are **no run time royalties**. Not even for the utilities.

Because the resulting system is truly multiuser.

Fast Programming is \$995 for a site license. VISA, MC, AMEX accepted. Currently available versions include PC-DOS, XENIX and UNIX. Call for specific machine availability.

Subject, Wills & Company

800 Enterprise Drive
Oak Brook, Illinois 60521
(312) 789-0240

The components of Fast Programming are:

C ROUTINES

Decimal math, keyin and display with windowing, extended string handling, time/date conversions, time/date edits, misc edits, sequential I/O, key sequential I/O, random I/O, indexed I/O, associative I/O, printer output, error handler, system interface routines and more.

C PROGRAM GENERATOR

Maintenance and data entry program generator, report program generator, processing program generator, I/O routines generator.

RUN TIME UTILITIES

AIMDEX, BUILD, CHAIN, COPY, CREATE, DELETE, DUMP, ENCODE, FILECHK, INDEX, LIST, REFORMAT and SORT.

MENU SYSTEM

Easy to use with no programming.

DOCUMENTATION GENERATOR

Create documentation fast.

The generators provide many user exits in the generated source code. This allows for custom coding without changing the generated C source program. This means that no custom coding is lost when programs are regenerated.