3

Chess Players, Music Lovers, and Mathematicians

In one inquiry it was found that a successful team of computer specialists included an ex-farmer, a former tabulating machine operator, an ex-key punch operator, a girl who had done secretarial work, a musician and a graduate in mathematics. The last was considered the least competent.

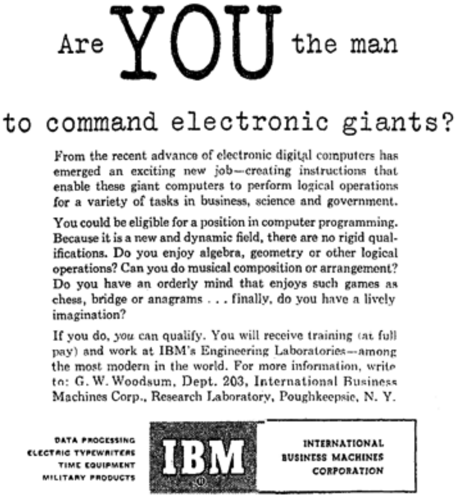
—Hans Albert Rhee, *Office Automation in a Social Perspective*, 1968

In Search of "Clever Fellows"

The "Talk of the Town" column in the *New Yorker* magazine is not generally known for its coverage of science and technology. But in January 1957, the highbrow gossip column provided for its readers an unusual but remarkably prescient glimpse into the future of electronic computing. Already there were more than fifteen hundred of the electronic "giants" scattered around the United States, noted the column editors, with many more expected to be installed in the near future. Each of these computers required between thirty and fifty programmers, the "clever fellows" whose job it was to "figure out the proper form for stating whatever problem a machine is expected to solve." And as there were currently only fifteen thousand professional computer programmers available worldwide, many more would have to be trained or recruited immediately. After expressing "modest astonishment" over the size of this strange new "profession we'd never heard of," the "Talk of the Town" went on, in its inimitable breezy style, to accurately describe a problem that industry observers were only just beginning to recognize: namely, that the looming shortage of computer programmers threatened to strangle in its cradle the nascent commercial computer industry.[1]

The impetus for the "Talk of the Town" vignette was a series of advertisements that the IBM Corporation had recently placed in the *New York Times*. At first glance the ads read as rather conventional help-wanted fare. Promising the usual “exciting new jobs” in a “new and dynamic field,” they sought out candidates for a series of positions in programming research. That particularly promising candidates might be those who “enjoy algebra, geometry and other logical operations” was also not remarkable, given the context. What caught the eye of the “Talk of the Town” columnists, however, was the curious addition of an appeal to candidates who enjoyed “musical composition and arrangement,” liked “chess, bridge or anagrams,” or simply possessed “a lively imagination.”[2] Struck by the incongruity between these seemingly different pools of potential applicants, one technical and the other artistic, the columnists themselves “made bold to apply” to the IBM manager in charge of programmer recruitment. “Not that we wanted a programming job, we told him; we just wondered if anyone else did.”[3]

The IBM manager they spoke to was Robert W. Bemer, a “fast-talking, sandy-haired man of about thirty-five,” who by virtue of his eight-years experience was already considered, in the fast-paced world of electronic computing, “an old man with a long beard.” It was from Bemer that they learned of the fifteen thousand existing computer programmers. An experienced programmer himself, Bemer nevertheless confessed astonishment at the unforeseen explosion into being of a programming profession, which even to him seemed to have “happened overnight.” And for the immediate future, at least, it appeared inevitable



Are YOU the man

to command electronic giants?

From the recent advance of electronic digital computers has emerged an exciting new job—creating instructions that enable these giant computers to perform logical operations for a variety of tasks in business, science and government.

You could be eligible for a position in computer programming. Because it is a new and dynamic field, there are no rigid qualifications. Do you enjoy algebra, geometry or other logical operations? Can you do musical composition or arrangement? Do you have an orderly mind that enjoys such games as chess, bridge or anagrams . . . finally, do you have a lively imagination?

If you do, *you* can qualify. You will receive training (at full pay) and work at IBM's Engineering Laboratories—among the most modern in the world. For more information, write to: G. W. Woodsum, Dept. 203, International Business Machines Corp., Research Laboratory, Poughkeepsie, N. Y.

Figure 3.1
IBM Advertisement, *New York Times*, May 31, 1969.

that the demand for programmers would only increase. With obvious enthusiasm, Bemer described a near future in which computers were much more than just scientific instruments, where "every major city in the country will have its community computer," and where citizens and businesspeople of all sorts—"grocers, doctors, lawyers"—would "all throw problems to the computer and will all have their problems solved." The key to achieving such a vision, of course, was the availability of diverse and well-written computer programs. Therein lay the rub for recruiters like Bemer: in response to the calls for computer programmers he had circulated in the *New York Times*, *Scientific American*, and the *Los Angeles Times*, he had received exactly seven replies. That IBM considered this an excellent return on its investment highlights the peculiar nature of the emerging programming profession.

Of the seven respondents to IBM's advertisements, five were experienced programmers lured away from competitors. This kind of poaching occurred regularly in the computer industry, and although this was no doubt a good thing from the point of view of these well-paid and highly mobile employees, it only exacerbated the recruitment and retention challenges faced by their employers. The other two were new trainees, only one of whom proved suitable in the long-term. The first was a chess player who was really "interested only in playing chess," and IBM soon "let him go back to his board." The second "knew almost nothing about computing," but allegedly had an IQ of 172, and according to Bemer, "he had the kind of mind we like. . . . [He] taught himself to play the piano when he was ten, working on the assumption that the note F was E. Claims he played that way for years. God knows what his neighbors went through, but you can see that it shows a nice independent talent for the systematic translation of values."[4]

Eventually the ad campaign and subsequent *New Yorker* coverage did net IBM additional promising programmer trainees, including an Oxford-trained crystallographer, an English PhD candidate from Columbia University, an ex-fashion model, a "proto-hippie," and numerous chess players, including Arthur Bisguier, the U.S. Open Chess champion, Alex Bernstein, a U.S. Collegiate champion, and Sid Noble, the self-proclaimed "chess champion of the French Riviera."[5] The only characteristics that these aspiring programmers appeared to have in common were their top scores on a series of standard puzzle-based aptitude tests, the ability to impress Bemer as being clever, and the chutzpah to respond to vague but intriguing help-wanted ads.

The haphazard manner in which IBM recruited its own top programmers, and the diverse character and backgrounds of them, reveals much about the state of computer programming at the end of its first decade of existence. On the one hand, computer programming had successfully emerged from the obscurity of its origins as low-status, feminized clerical work to become the nation's fastest-growing and highest-paid technological occupation.[6] The availability of strong programming talent was increasingly recognized as essential to the success of any corporate computerization effort, and individual programmers were able to exert an inordinate amount of control over the course of such attempts.

But at the same time, the "long-haired programming priesthood"—the motley crew of chess players, music lovers, and mathematicians who comprised the programming profession in this period—fit uncomfortably into the traditional power structures of the modern corporate organization.[7] The same arcane and idiosyncratic abilities that made them well-paid and highly sought-after individuals also made them slightly suspect. How could the artistic sensibilities and artisanal practices of programmers be reconciled with the rigid demands of corporate rationality? How could corporate managers predict and control the course of computerization efforts when they were so dependent on specific individuals? If good programmers "were born, not made," as was widely believed, then how could the industry ensure an adequate supply?[8]

The tension between art and science inherent in contemporary programming practices, unwittingly but ably captured by the "Talk of the Town" gossip columnists, would drive many of the most significant organizational, technological, and professional developments in the history of computing over the course of the next few decades. This chapter will deal with early attempts to use aptitude tests and personality profiles to manage the growing "crisis" of programmer training and recruitment.

The Persistent Personnel Problem

The commercial computer industry came of age in the 1960s. At the beginning of that decade the electronic computer was still a scientific curiosity, its use largely confined to government agencies as well as a few adventurous and technically sophisticated corporations; by the decade's end, the computer had been successfully reinvented as a mainstream business technology, and companies such as IBM, Remington Rand, and Honeywell were selling them by the thousands.