Methods of Teaching Exact and Natural Sciences Using Computer Technologies to Agricultural Students

Abdusamatova Hilola Odiljon Qizi,
Shamishova Aysholpan Tejetdinovna,
Xudoyberdieva Mavluda Ibrayimovna,
Abdurahmonov Dilmurod Egamberdiyevich,
Ochilov Asadullo Izzatulloevich

Abstract--- *The article deals with the new computer technologies in the agricultural lesson. Today we are living in technology century. We can not imagine our life without computer and internet.*

Keywords--- *Computer, Language, Technology, Useful, Century, Great, Skill.*

I. INTRODUCTION

With the rise of the World Wide Web came the notion of ‘‘Internet Time.’’ Netscape’s founder Jim Clark called it ‘‘Netscape Time’’ in his 1999 book by that title: he defined it as a telescoping of the time for a technology to proceed from invention to prototype, production, commercial success, maturity, and senescence.1 The historian faces a modern version of Zeno’s paradox. In the classical story, a fast runner never reached the finish line in a race, because he first had to traverse one-half the distance to the end, which took a finite time, and then one half the remaining distance, which again took a smaller but still finite time, and so on. There is a finite time between sending a completed manuscript to the typesetter and the delivery of a book or journal article to the reader. When the subject is computing, Zeno’s paradox takes control: enough happens in that brief interval to render what was just written obsolete. Many recognize this and embrace the solution of publishing electronically, thus telescoping that time down to zero.

There are indeed many Web sites devoted to the history of computing, some of excellent quality. Still, embracing Web publishing is a false hope, because it does nothing to compress the time spent organizing historical material into a coherent narrative. History is a chronology of facts, but the word history contains the word story in it, and telling stories is not rendered obsolete by technology. The storyteller neither can, nor should, speed that activity up.

In looking over the first edition, I feel that it has managed to avoid Zeno’s trap. A number of significant events have developed after 1995, and in a new chapter I examine three at length. These are the Microsoft trial, mentioned above; the explosion and equally stunning implosion of the ‘‘dot.com’’ companies; and the rise of the ‘‘open

Abdusamatova Hilola Odiljon Qizi, Teacher, Termez Branch of Tashkent State Agrarian University.
Shamishova Aysholpan Tejetdinovna, Teacher, Termez Branch of Tashkent State Agrarian University.
Xudoyberdieva Mavluda Ibrayimovna, Teacher, Termez Branch of Tashkent State Agrarian University.
Abdurahmonov Dilmurod Egamberdiyevich, Teacher, Termez Branch of Tashkent State Agrarian University.
Ochilov Asadullo Izzatulloevich, Teacher, Termez Branch of Tashkent State Agrarian University.

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020

source'' software movement and especially the adoption of the Linux operating system. These are three of at least a dozen topics that I could have chosen, but to examine more would not serve the reader.

Zeno may get his revenge yet. The above plan for bringing the history of computing up to date seems rational, but it may have a fatal flaw. The history of computing, as a separate subject, may itself become irrelevant. There is no shortage of evidence to suggest this. For example, when the financial press refers to ''technology'' stocks, it no longer means the computer industry represented by companies like IBM or even Intel, but increasingly Internet and elecommunications firms. In my work as a museum curator, I have had to grapple with issues of how to present the story of computing, using artifacts, to a public. It was hard enough when the problem was that computers were rectangular ''black boxes'' that revealed little of their function; now the story seems to be all about ''cyberspace,'' which by definition has no tangible nature to it.

Perhaps the invention of the computer is like Nicholaus Otto's invention of the four-cycle gasoline engine in 1876. However significant that was, if Otto is remembered at all it is because the Otto Cycle became the preferred way to power the automobile. And the automobile in turn is a topic worthy of study not so much for its intrinsic qualities as a machine, but for helping shape a society that has grown around personal transportation. In the preface to the first edition I suggested that this book's emphasis on the transition from batch-oriented to interactive computing might some day seem to be a minor part of computing history. Has that day come already? What now seems to have been critical was the transformation of the computer from a stand-alone to a networked device. That, however, could not have happened were it not for the earlier transition from batch to interactive use. Although the hardware roots of cyberspace are found in chip manufacturers including Intel, and in personal computer companies like Apple, the spiritual roots of cyberspace are found in time-sharing experiments like Project MAC. I do not feel that the history of computing will vanish into a subfield of the history of cyberspace. The recent implosion of the dot.com companies (the second topic covered in the new chapter) suggests that a study of hardware and software (including Linux, the third topic) will remain at the core of any history. The study of cyberspace is merging with social, cultural, military, and political history, as digital technologies increasingly mediate among human interactions. That is the origin of the term media. I hope this book will continue to serve those who wish to know how the increasingly mediated world we now live in arose.

II. METHODS

Grammar- Translate method is used in this article. Computers were invented to ''compute'': to solve ''complex mathematical problems,'' as the dictionary still defines that word.1 They still do that, but that is not why we are living in an ''Information Age.'' That reflects other things that computers do: store and retrieve data, manage networks of communications, process text, generate and manipulate images and sounds, fly air and space craft, and so on. Deep inside a computer are circuits that do those things by transforming them into a mathematical language. But most of us never see the equations, and few of us would understand them if we did. Most of us, nevertheless, participate in this digital culture, whether by using an ATM card, composing and printing an office newsletter, calling a mail-order house on a toll-free number and ordering some clothes for next-day delivery, or shopping at a mega-mall where the inventory is replenished ''just-in-time.'' For these and many other applications, we can use all

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020

the power of this invention without ever seeing an equation. As far as the public face is concerned, ‘‘computing’’ is the least important thing that computers do. But it was to solve equations that the electronic digital computer was invented. The word ‘‘computer’’ originally meant a person who solved equations; it was only around 1945 that the name was carried over to machinery [2]. That an invention should find a place in society unforeseen by its inventors is not surprising.3 The story of the computer illustrates that. It is not that the computer ended up not being used for calculation—it is used for calculation by most practicing scientists and engineers today.

That much, at least, the computer’s inventors predicted. But people found ways to get the invention to do a lot more. How they did that, transforming the mathematical engines of the 1940s to the networked information appliance of the 1990s, is the subject of this article. Yet who would deny that computing technology has been anything short of revolutionary? A simple measure of the computing abilities of modern machines reveals a rate of advance not matched by other technologies, ancient or modern. The number of computers installed in homes and offices in the United States shows a similar rate of growth, and it is not slowing down. Modern commercial air travel, tax collection, medical administration and research, military planning and operations— these and a host of other activities bear the stamp of computer support, without which they would either look quite different or not be performed at all. The history of computing commands—as it probably should—more attention from the public than the history of the washing machine. The colleague who in 1981 dismissed the study of computing no longer prepares his papers on a manual typewriter, I suspect. Historians are among the most fanatic in embracing the latest advances in computer-based aids to scholarship [7]. Is the electronic computer only one of many large-scale, high-technology systems that have shaped the twentieth century? To what extent is it unique as an information-processing machine? To what extent is computing after 1945 different from the information-handling activities of an earlier age? The popular literature tends to stress computing’s uniqueness, hand in hand with breathless accounts of its revolutionary impacts. Some writers cast this revolution as a takeover by a ‘‘clean’’ technology, with none of the pollution or other side effects of the technologies of the Iron Age.8 If the computer is revolutionizing our lives, who is on the losing side; who are the loyalists that computing must banish from this new world? Or is computing like the ruling party of Mexico: a permanent, benign, institutionalized ‘‘revolution’’?

III. RESULTS AND DISCUSSION

Current studies of computing give conflicting answers to these questions.

Some show the many connections between modern computing and the information-handling machinery and social environments that preceded it [1]. Some make passing references to computing as one of many technologies that owe their origins to World War II research. Many stress the distinction between computing and other products of wartime weapons laboratories; few examine what they have in common[11]. Still others make little attempt to discover any connection at all. In writing about the emergence of electrical power systems in the

United States and Europe, Thomas Parke Hughes introduced the notion of technological systems, into which specific pieces of machinery must fit. His work is too rich and complex to be summarized here, but a few aspects are particularly relevant to the history of computing. One is that ‘‘inventors’’ include people who innovate in social, political, and economic, as well as in technical, arenas. Sometimes the inventor of a piece of hardware is also the

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020

pioneer in these other arenas, and sometimes not. Again and again in the history of computing, especially in discussing the rise of Silicon Valley, we shall encounter an entrepreneur with a complex relationship to a technical innovator. This narrative will also draw on another of Hughes's insights: that technology advances along a broad front, not along a linear path, in spite of terms like ‘‘milestone’’ that are often used to describe it. The history of computing presents problems under this systems approach, however. One definition of a modern computer is that it is a system: an arrangement of hardware and software in hierarchical layers. Those who work with the system at one level do not see or care about what is happening at other levels. The highest levels are made up of ‘‘software’’—by definition things that have no tangible form but are best described as methods of organization. Therefore, it might be argued, one need not make any special effort to apply the systems approach to the history of computing, since systems will naturally appear everywhere. This is another example of computing's uniqueness. Nevertheless, the systems approach will be applied in this narrative, because it helps us get away from the view of computing solely as a product of inventors working in a purely technical arena. Another approach to the history of technology is known as ‘‘social construction.’’ Like the systems approach, it is too rich a subject to be summarized here [14]. Briefly a social constructionist approach to the history of computing would emphasize that there is no ‘‘best’’ way to design computing systems or to integrate them into social networks. What emerges as a stable configuration—say, the current use of desktop systems and their software—is as much the result of social and political negotiation among a variety of groups (including engineers) as it is the natural emergence of the most efficient or technically best design. A few historians of computing have adopted this approach, but most have not, preferring to describe computing's history as a series of technical problems met by eng00ineering solutions that in hindsight seem natural and obvious. However, a body of historical literature that has grown around the more recent history of computing does adopt a social constructionist approach, if only informally. The emergence of personal computing has been the subject of popular books and articles by writers who are either unfamiliar with academic debates about social construction or who know of it but avoid presenting the theory to a lay audience. Their stories of the personal computer emphasize the idealistic aspirations of young people, mainly centered in the San Francisco Bay area and imbued with the values of the Berkeley Free Speech Movement of the late 1960s. For these writers, the personal computer came not so much from the engineer's workbench as from sessions of the Homebrew Computer Club between 1975 and 1977. These histories tend to ignore advances in fields such as solid state electronics, where technical matters, along with a different set of social forces, played a significant role. They also do little to incorporate the role of the U.S. Defense Department and NASA (two of the largest employers in Silicon Valley) in shaping the technology. These federal agencies represent social and political, not engineering, drivers. I shall draw on Hughes's concepts of social construction and his systems approach throughout the following narrative; and we will find abundant evidence of social forces at work, not only during the era of personal computing but before and after it as well. Local networking took the ‘‘personal’’ out of personal computing, at least in the office environment. (One could still do whatever one wanted at home.) PC users in the workplace accepted this Faustian bargain. The more computer-savvy among them resisted, but the majority of office workers hardly even noticed how much this represented a shift away from the forces that drove the invention of the personal computer in the first place. The ease with which this transition took place shows that those who believed in truly autonomous, personal computing were perhaps naive. Still, the

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020

networked office computers of the 1990s gave their users a lot more autonomy and independence than the timeshared mainframes accessed through ‘‘dumb terminals’’ or ‘‘glass Teletypes’’ in the 1970s. It was just not how the people at Byte magazine or the Homebrew Computer Club had imagined things would evolve.

Most benefits of connecting office workers to a LAN went to administrators and managers. For their part, users no longer had to worry about backing up files—something few PC owners ever learned to do faithfully anyway—and they could now exchange files and messages with one another using electronic mail. But there was one unanticipated, very important thing that users connected to a LAN got in return—access to the Internet. The present-day Internet, though well known, is hard to define. It is descended from the ARPANET describe. Like ARPANET and the other networks described earlier, the Internet uses ‘‘packet switching.’’ Sending a message does not require a dedicated connection from one computer to another, as, say, one has when calling someone on the telephone. There are however several major differences. The Internet is not a single network but rather the connection of many different networks across the globe; hence the name. Some of those networks are open to the public, not just to a restricted or privileged community. (Note there are still many networks that are restricted, e.g., one used by a bank for its internal operations.) Finally, the Internet allows communication across these different networks by its use of a common protocol, TCP=IP (transmission control protocol= internet protocol). This interconnection of networks to one another, using the glue of TCP=IP, constitutes the present-day Internet. The Internet made its way into general use by a combination of social and technical factors. Among the former was the shift of financial and administrative support from ARPA, to the National Science Foundation connected to mainframes by time-sharing. (As the ARPANET took shape one could also, in a few places, connect a terminal directly to the network through a terminal interface processor (TIP)—a variation of the IMP concept.) With the invention of Ethernet in 1973, and the personal computer the following year, the economics of computing changed. Computing power was no longer scarce. Time-sharing matured and became available on many mainframes, but it was supplanted by client-server computing that descended from the work at Xerox-PARC. And throughout this era Moore’s Law ruled: computing power, as measured by the density of the silicon chips that went into these machines, was doubling about every eighteen months.

Local area networks made it possible for large numbers of people to gain access to the Internet. Ethernet’s speeds were fast enough to match the high speeds of the dedicated lines that formed the Internet’s backbone. High-speed networking had always been among the features workstation companies wanted to supply—recall SUN’s marketing slogan: ‘‘The Network is the Computer.’’ What had not been anticipated was how advances in personal computers, driven by ever more powerful processors from Intel, brought that capability to offices and other places outside the academic and research worlds. By the late 1980s those with UNIX workstations, and by 1995 those with personal computers on a LAN, all had access to the Internet, without each machine requiring a direct connection to the Internet’s high-speed lines. Ethernet’s high data rates thus provided a way of getting around the fact that communication speeds and data capacity had not kept up with the advances in computer processing speeds and storage. Gordon Moore’s colleague at Intel, Andrew Grove, came up with his own ‘‘law’’ to describe this disparity: while chip density doubles every eighteen months (Moore’s Law), telecommunications bandwidth doubles every 100 years (Grove’s Law).37 Bandwidth growth has picked up since 1990, but it still lags well behind the growth of

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020

chip density. Grove believes the disparity is a result of overregulation of the telecommunications industry. Whatever the cause, it is true that the telephone lines coming into homes and offices cannot handle data at more than about 50 thousand bits per second—usually less, and in any case well below the speeds achieved by Ethernet and required by many Internet activities. Since the mid-1990s modem manufacturers have made heroic efforts to improve data rates for ordinary telephone connections. Plans have also emerged to use other wires that come into the home for Internet traffic: the line that carries cable television or even the power lines. Various satellite or microwave wireless technologies are also being developed. One of these approaches will probably break the bottleneck. Meanwhile, people do connect their home computers to the Internet by dialing a local telephone number, but the access they get is a fraction of what they can find at the office, laboratory, or university. For now, Grove's Law seems to hold.

Networking III: The World Wide Web

As the Internet emerged from its roots in ARPA, it began to change. The initial activities on the Internet were ARPANET derived: users could log on to a remote computer, transfer large files from one machine to another, and send mail. The first two activities later known as (‘‘Telnet’’ and ‘‘FTP’’) were explicit goals of the original ARPANET; mail was not, but it emerged soon after the first few nodes were working. Early e-mail facilities were grafted onto the file-transfer operation, but before long dedicated e-mail software was developed, with most of the features found in modern e-mail systems already in place (e.g. the ability to reply to someone, save a message, or send a message to a list).38 The first serious extension to that triad gave a hint of what the popular press calls a ‘‘virtual community’’ based on the Internet. Whether that phrase has any meaning, and if so, what it is, will be discussed later, but what evoked it was the development of news or discussion groups on the early Internet. Although these groups are associated with the Internet, for years only those with access to UNIX systems had access to them. For the general public they were anticipated in the personal computer arena by so-called bulletin-board systems (BBSs), which as the name implies, acted like bulletin boards, on which anyone could post a note for all to read. BBSs typically ran on limited facilities, using public-domain software running on an IBM XT or the equivalent. Key technical developments were the introduction in 1981 of an inexpensive modem by Hayes Microcomputer Products, and of the XT itself in 1983, with its 10- megabyte hard disk. Users dialed into these bulletin boards with a local telephone call, at a rate of a few hundred bits per second.39 But they worked and were well liked, and some remained in use into the 1990s. UNIX-based news groups first appeared after 1979, somewhat independently of the mainstream ARPANET-Internet activities and under the general name of Usenet.40 These were arranged into a set of major, for example, ‘‘comp’’ for computers, or ‘‘rec’’ for hobbies. The narrative in the first edition ended on August 8, 1995, the day that Netscape offered shares on the stock market. The commercialization of the Internet, and the role that Netscape played in it, ushered in a new era in computing. It is too early to write a history of this era. There is no clear theoretical framework on which the historian can build a narrative. Still, so much has happened in the past few years that one cannot put off an attempt to write some kind of historical narrative about the ‘‘dot.com’’ phenomenon. A section of this chapter will do that, but this chronicle of the inflation and bursting of the dot.com bubble is very much a work in progress. This chapter also addresses two other developments of the past few years. Like the dot.com phenomenon, these are ongoing developments whose direction seems to change daily if one reads the newspaper headlines. Fortunately, these developments have nice connections to events

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020

of computing's ‘‘ancient history’’ (i.e., before 1995). Thus they allow the historian to gain a glimmer of perspective. The antitrust trial against Microsoft, discussed first, is the culmination of a sequence of legal actions taken against the company, and it reflects issues that were present at Microsoft as early as 1975, when the company was founded. Not only that, the Microsoft trial echoes many of the arguments made against IBM during its legal troubles with the U.S. Justice Department in the 1970s. The discussion of the GNU/Linux operating system and the ‘‘open source’’ software movement, discussed last, likewise has deep roots as well as the controversy over who was allowed to use and modify the TRAC programming language. GNU/Linux is a variant of UNIX, a system developed in the late 1960s and discussed at length in several earlier chapters of this book. UNIX was an open system almost from the start, although not quite in A commercial, aired during the third quarter of the game, was the most memorable part of the broadcast of the January 1984 Super Bowl. The Macintosh, Apple assured us, would usher in a new era of personal computing, and therefore the year 1984 would not be one of dreary conformity and oppression as prophesied by George Orwell's novel 1984. A revolution in personal computing was indeed in the works, and the Macintosh was leading the way. But Microsoft, not Apple, helped bring the revolution to a mass market. That happened not in 1984, the year the Mac appeared, but in 1992, when Microsoft began shipping version 3.1 of its Windows program. In 1984, Apple hoped that the Mac would bring the innovative ideas from the Xerox Palo Alto Research Center, ideas already present in a few personal computer systems, to the consumer. A dozen years later, Microsoft, not Apple, would dominate personal computer software.1 And that domination, in turn, would lead to its entanglement in a bitter antitrust trial. Just as IBM spent a significant fraction of its resources during the 1970s facing a challenge by the U.S. Justice Department, so too is Microsoft in the same situation, following a similar filing against it in 1997. In November 2001 the federal government announced a settlement, but several states, and the European Union, refused to go along. Their arguments were also rejected by a ruling on November 1, 2002. Almost daily, the business press reports whenever a judge or lawyer makes a statement. Until the case is settled, one can only make provisional comments about its significance. The lesson of the IBM trial, however, applies to the present case against Microsoft: namely that the Justice Department is not a place that recognizes how advancing technology will render much of the lawsuit irrelevant. What is the Microsoft-equivalent of the personal computer, whose appearance in the midst of the IBM trial was ignored as the litigants fought over mainframe dominance? It is too early to tell, although I will discuss some candidates later in this chapter. What is certain is that advances in computing already threaten, and will continue to threaten, Microsoft's ability to dominate personal computing, based on its Windows and Office software. The licensing policies of Microsoft and Intel gave rise to clone manufacturers, like Dell, Compaq, and Gateway, who provided choices unavailable to Apple customers. (Apple, for most of its history, has refused to license its Macintosh software to third-party computer makers.) That policy yielded a greater variety of products and, above all, lower prices for computers based on Intel microprocessors and running Microsoft's DOS and then Windows. Windows version 3.1, Intel's introduction of the Pentium processor, and Microsoft's combining applications software into a suite called Microsoft Office, combined to give consumers, let's say, 80 percent of what the Macintosh was offering, at a lower price for the total package. To Apple's surprise (and to the chagrin of Mac fans), that percentage was good enough to tip the balance, perhaps forever, away from Apple. By 1995 the advantage of Apple's more elegant design no longer mattered, as the Microsoft/Intel combination

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020

became a standard, like COBOL in the 1960s. As with COBOL, what mattered was the very existence of a standard, not the intrinsic value or lack thereof of the software. The Macintosh Connection One could begin this story of Microsoft's triumph and troubles.

IV. CONCLUSION

People never did succeed in building gallium arsenide circuits that could compete with silicon, even if people did learn how to pronounce the term. Linux, however it is pronounced, is going to have to deal with Microsoft one way or another. The experience with Java shows that simply being an alternative to Microsoft is not sufficient in itself to prevail. Among Linux evangelists are a strong and vocal group who tout Linux-based programs that offer a graphical interface like Windows (''KDE'' and ''GNOME''), word processors (''AbiWord''), and other products.86 In keeping with the UNIX philosophy, and in contrast to Windows, the code that generates the graphical user interfaces is kept separate from the base Linux code. Linux is still accessed by typing a command line, like DOS. As Microsoft moved away from DOS, Linux enthusiasts steadfastly prefer typing cryptic commands, many of which resemble the DOS commands of old. Apple seems to be of two minds on this. When it introduced the Macintosh in 1984 it got rid of the command line, but with the latest version of the Mac operating system (''X,'' based on UNIX), a savvy user can bypass the graphical interface that Apple made so famous.

So using computer technology is important in today`s our lifes and bringing up children to fight against their power.

REFERENCES

[1] Adams Associates. ''Computer Characteristics Chart.'' *Datamation* (November/December 2010): 14–17.
[2] Adams, Charles W., and J. H. Laning Jr. ''The MIT Systems of Automatic Coding: Comprehensive, Summer Session, and Algebraic.'' *Symposium on Automatic Programming of Digital Computers* (2004): 40–68.
[3] Aiken, Howard. ''The Future of Automatic Computing Machinery.'' *NTF-4 Special Edition* (2000).
[4] Allen, John. ''The TX-0: Its Past and Present.'' *Computer Museum Report*, 8 (spring 1984): 2–11.
[5] Allison, David K. ''U.S. Navy Research and Development Since World War II.'' *In Military Enterprise and Technological Change, edited by Merritt Roe Smith*, 289–328. Cambridge: MIT Press, 1985.
[6] 6. Anderson, Christopher. ''The Rocky Road to a Data Highway.'' *Science* 260 (May 21, 1993): 1064–1065.
[7] Aspray, William. John von Neumann and the Origins of Modern Computing. *Cambridge: MIT Press,* 1990.
[8] Aspray, William, ed. Computing Before Computers. *Ames, Iowa: Iowa State University Press,* 1990.
[9] Aspray, William, ed. Technological Competitiveness. *New York: IEEE Press,* 1993.
[10] Ante, Spencer E. ''Big Blue's Big Bet on Free Software.'' *Business Week* (December 10, 2001): 78–79.
[11] Backus, John. ''Programming in America in the 1950s—Some Personal Recollections.'' In A History of Computing in the Twentieth Century, 125–135. *New York: Academic Press,* 1980.
[12] Baecker, Ronald, and William A. S. Buxton. Readings in Human-Computer Interaction: A Multidisciplinary Approach. *Los Altos, CA: Morgan Kaufmann,* 1987.
[13] Banks, David. Breaking Windows: How Bill Gates Fumbled the Future of Microsoft. *New York: Free Press,* 2001.
[14] Barden, William Jr. How to Buy and Use Minicomputers and Microcomputers. *Indianapolis, IN: Howard Sams,* 1976.
[15] Basalla, George. ''The Evolution of Technology.'' *Cambridge, UK: Cambridge University Press,* 2008.
[16] Bashe, Charles J., Lyle R. Johnson, John H. Palmer, and Emerson W. Pugh, IBM's Early Computers. *Cambridge: MIT Press,* 2000.

DOI: 10.37200/IJPR/V24I5/PR2020779
Received: 23 Apr 2020 | Revised: 30 Apr 2020 | Accepted: 24 May 2020