

DOI:10.1145/2804228

Thomas Haigh and Mark Priestley

Historical Reflections Innovators Assemble: Ada Lovelace, Walter Isaacson, and the Superheroines of Computing

Can computing history be both inspiring and accurate?

onsider two recent blockbuster sequels. Avengers: Age of Ultron, a superhero movie, enjoyed the second strongest opening weekend of all time, behind only its predecessor, Avengers Assemble. The fastest-selling history of computing book ever published is Walter Isaacson's The Innovators: How a Group of Hackers, Geniuses, and Geeks Created the Digital Revolution. Its sales fall short only in comparison to his previous book, Steve Jobs, which reportedly broke all records for a biography.

Avenging and innovating turn out to have a surprising amount in common. Both require one to assemble a team of superheroes who must work together to defy daunting odds and change the course of human history. Both deploy a cast of characters who have been written about for decades but are now reaching massive audiences. Both feel somewhat overstuffed, as their hugely experienced creators struggle to maintain a light touch while maneuvering a complicated narrative through a vast number of required plot points. Both highlight origin stories, as if understanding the moments at which individuals received their special powers or the circumstances in which particular technologies were first coaxed into operation will always explain their subsequent trajectories.

Isaacson's geek revolutionaries are, for the most part, entrepreneurs rather than academics. People are interested in the men behind the companies behind the gizmos of daily life, particularly if those men became spectacularly rich while exhibiting captivating flaws. Hence the wealth of books and films about Steve Jobs, Bill Gates, Mark Zuckerberg, and the early days of Google. Most of the computer science students featured in these stories dropped out part way through their degrees. Computer science has invested little effort in building and celebrating its own set of heroic role models. Individuals such as Edsger Dijkstra, Donald Knuth, and Alan Kay all have their followings but none have yet inspired a full-length biography, a statue, or a museum. Even John von Neumann has largely slipped from general awareness in the decades since his death. Perhaps computer scientists feel their discipline is doing pretty well without devoting significant energy to the construction and celebration of male heroes. Alan Turing is the exception that proves the rule here, given his gripping personal story, significant contribution to the Second World War, and crossover appeal as a gay martyr.

Isaacson, who has headed both CNN and *Time* Magazine, is one of the world's most successful and best-connected journalists. His titular promises of "geeks" and "genius" signal this is a fairly conservative retelling of computer history, discussing the invention of technologies rather than their commercialization or use. He arranges a series of vignettes along a familiar arc, leading from the Victorian dreams of Charles Babbage through the various computer inventors of the 1940s to the networking pioneers of the 1960s, the home computer hackers of the 1970s, and a variety of Internet entrepreneurs. Isaacson interviewed famous people to enliven the more recent history, but in his earlier chapters he mines the work of historians.

There are two areas in which Isaacson departs from the classic storyline of popular computing history. Like the Avengers movies, his book suggests superheroes need to collaborate to get the job done. We historians of science and technology often complain that nobody reads our scholarly tomes. Usually we blame this on the unshakable attachment of the public to books about "the lone genius who revolutionized" something or other. Isaacson's subtitle confirms he has no problem with the "genius" part of that, but is rightly critical of the silly idea that his "hackers, geniuses, and geeks" are most innovative when deprived of resources and forced to work alone.

The other distinctive feature of Isaacson's book is his stress on the contribution of women to innovation in computing. He foregrounds a trio of coding superwomen, and centered the promotional campaign for the book on the claim that he was avenging brilliant but defenseless women who had been unjustly erased from history. Avenging is less hospitable to women. Age of Ultron was criticized for marginalizing its female characters, despite being directed by Joss Whedon whose Buffy the Vampire Slayer broke new ground with its stake-happy heroine and her lesbian witch sidekick. Disney's failure to produce an action figure of Black Widow, the only female Avenger, provoked a chorus of angry tweets.

According to Isaacson, men were good at hardware, but only women were smart enough to appreciate the need for software. The idea that women invented programming is not entirely novel, having been repeated widely enough in recent years to have become the new conventional wisdom. As we write, each of the top 10 hits returned by Googling "first computer programmer" relates to one of three topics. Hits number 1, 4, 6, 7, 8, 9, and 10 are about Ada Lovelace. Number 2 is about Grace Hopper, while number 3 is about the "women of ENIAC." Number 5 is "The Forgotten Female Programmers Who Created Modern Tech," a lengthy inter-



Ada Lovelace.

view with Isaacson on NPR's flagship "Morning Edition" radio program in which he hits the trifecta.

Ada Lovelace

One might reasonably question whether Lovelace needed Isaacson to rescue her from obscurity. He first learned of Lovelace in 2007 when his teenage daughter wrote a college essay about her as the family summered in Aspen, apparently inspired by a "Batman comic book."^a Isaacson might not have heard of Lovelace, but she was already more famous than any Turing Award winner, having inspired at least a dozen biographers, a major computer language, several novels, a Google Doodle, a comic book series, and a movie. Launched in 2009 by social media consultant Suw Charman-Anderson, Ada Lovelace Day has spread quickly to encompass a host of worldwide events intended to celebrate "the achievements of women in science, technology, engineering, and maths." Lovelace has become a superhero for all of STEM, not just for computing.

a http://nyti.ms/1vAolLZ

Lovelace is remembered for her collaboration with Charles Babbage to publicize the mechanical calculators he designed and attempted to build from the 1820s onward. Babbage relied substantially on the writings of others to promote his machines. The first publication describing the Analytical Engine, intended to automatically execute a sequence of mathematical operations, was an 1842 article by the Italian Luigi Menabrea, written after Babbage's visit to Turin in 1840. Lovelace translated Menabrea's French-language original, adding seven explanatory "Notes" that were more than twice the length of Menabrea's original text.^b These included a detailed discussion of how to use the Analytical Engine to calculate the Bernoulli numbers. This work, summarized in a large table describing the calculation, has long been understood as a computer program avant la lettre.^c That made Lovelace "the first programmer," an identification cemented in the 1980s when the language ADA was named in her honor.

Google's N-gram tool, based on a massive full-text database of Englishlanguage books, suggests that by the mid-1990s Lovelace's fame had already

Figure 1. Google's N-gram tool view: Howard Aiken, Presper Eckert, Grace Hopper, Ada Lovelace, and Alan Turing.



outstripped those of computer-builders such as Presper Eckert and Howard Aiken. Alan Turing, whose story is similarly romantic, enjoyed a still sharper climb in public awareness (see Figure 1).

Lovelace has been celebrated as much for her visionary asides as her mathematical accomplishments. She imagined a version of the Analytical Engine that would allow mechanical representation of "the fundamental relations of pitched sounds in the science of harmony" and "compose elaborate and scientific pieces of music." These remarks carry an undeniable frisson of prescience, leading science writer and television presenter Steven Johnson to the idea that Lovelace was a "time travelling" inventor who, like Leonardo da Vinci and Charles Darwin, somehow existed outside her own time. According to Johnson, "her footnote opened up a conceptual space" eventually filled by "Google queries, electronic music, iTunes, hypertext, Pixar."d One recent book, A Female Genius: How Ada Lovelace, Lord Byron's Daughter, Started the Computer Age, claims that Lovelace "foresaw the digitization of music as CDs."8 In a promotional interview for the fashion section of the New York Times. Isaacson claimed "Ada Lovelace defined the

digital age," arguing with a wave that seemed to "encompass all of Silicon Valley and the techies sitting around us" that "If it wasn't for Ada Lovelace, there's a chance that none of this would even exist." This hand waving is not untypical, as is the failure to distinguish between imagining something and causing that thing to come about.

Isaacson acknowledges that the extent of Lovelace's contribution to Babbage's overall project has been energetically debated. Researchers have minutely parsed the surviving records to build or challenge the case for Lovelace as a mathematical genius or as an essential contributor to Babbage's project. The progress of work on the Bernoulli example can be followed in the correspondence between the pair, complicating the popular idea of a neat division of labor in which Lovelace worked out how to use the hardware Babbage invented. Isaacson steers a middle course here, and as a result has been criticized for suggesting Lovelace was "never the great mathematician that her canonizers claim," thus denying her admittance to the club of "genius" promised in his subtitle and perhaps undercutting his own vague claims for her significance.^e

We feel this squabbling over authorship misses the bigger point. What does it mean for Isaacson to call Lovelace an essential precondition to the existence of today's tech world?

b "Sketch of the Analytical Engine invented by Charles Babbage Esq. By L.F. Menabrea, of Turin, officer of the Military Engineers, with notes upon the memoir by the translator," *Taylor's Scientific Memoirs*, Vol. 3 (1843). Menabrea's original article is reprinted alongside Lovelace's translation and notes in *The Works of Charles Babbage, Volume 3: The Analytical Engine and Mechanical Notation*, M. Campbell-Kelly, Ed. (William Pickering, London, 1989); see http://bit.ly/18FbNZL.

For example, in one of the earliest popular accounts of computers, Faster Than Thought (B.V. Bowden, Ed., Pitman, 1953). This book included a reprint of the text of the Notes, and its frontispiece was a portrait of Lovelace herself. In 1984, Dorothy Stein reproduced part of the table with the caption "The diagram or program for the computation of the first seven Bernoulli numbers" (D.K. Stein, "Lady Lovelace's Notes: Technical Text and Cultural Context," Victorian Studies 28, 1 (1984)), and according to Benjamin Woolley, "It is this table which is used to justify the claim that Ada was the world's first computer programmer" (B. Woolley, The Bride of Science: Romance, Reason and Byron's Daughter (Macmillan, 1999)). Isaacson tells us that Lovelace "figure[d] out in step-by-step detail the workings of what we now call a computer program or algorithm ... described a sequence of operations and then made a chart showing how each would be coded into the machine."

d Johnson, S. The Tech Innovators of the Victorian Age: What the Victorian Computing Pioneers Can Teach Us About Invention and Time Travelling. *Financial Times*, (Oct. 17, 2014), showcasing material from his 2014 book *How We Got to Now* (Riverhead Books).

e http://bit.ly/1J4hHSh

Logically, it means those who created later technologies drew directly on her work with Babbage, and indeed that without them nobody would ever have thought of programming a computer. For example, Isaacson explained during his NPR interview that "Babbage's machine was never built. But his designs and Lovelace's notes were read by people building the first computer a century later." However, expert historians have been unable to find any substantial influence of Babbage on the designers of early computers.

Isaccson's claim most likely refers to Howard Aiken, a Harvard University computer pioneer who rhetorically enlisted Babbage as kind of patron saint for his project, since to the best of our knowledge no claim has ever been made that Babbage or Lovelace influenced others such as Konrad Zuse. Tommy Flowers, John V. Atanasoff, J. Presper Eckert, or John W. Mauchly. I. Bernard Cohen, who looked more closely than any other historian at Aiken's work, concluded he became aware of Babbage only after proposing to build his computer and did not learn of Lovelace's work or appreciate the importance of conditional branching for some years later. Cohen discovered that "Aiken was generally ignorant of Babbage's machines" when writing his 1937 proposal to construct what became the Harvard Mark I, having seen only brief accounts, and including "summary (and somewhat erroneous) statements about Babbage's machines." "Aiken," wrote Cohen, "praised Babbage to enhance his own stature" even though "Babbage did not play a major role in the development of Aiken's ideas." Cohen quotes Grace Hopper, who worked closely with Aiken to oversee programming of his computer, as saying she was unaware of Lovelace's work "until 10 or 15 years later."5

Turning to the specifics of Lovelace's notes, we challenge the common characterization of Lovelace's tabular presentation of the Bernoulli calculation as a "program." It is not clear what would constitute a program for a computer that was never fully designed, still less built, but neither Lovelace nor Menabrea (whose paper included several smaller examples reportedly supplied to him by Babbage¹⁶) ever produced a description of a calculation in a form detailed enough to be processed directly by the Analytical Engine. Babbage intended his computer to read and execute instructions one at a time from a deck of "operation cards." The famous table is not a specification for a card deck to compute the Bernoulli numbers. Instead, as Lovelace explained, it "presents a complete simultaneous view of all the successive changes" the various storage units of the Analytical Engine would "pass through" in computing the specific number B₇. When computing other Bernoulli numbers the engine would carry out different sequences of operations, leading to tables with more or fewer rows. The control logic to produce these variations is not present in the table itself. In modern terminology, the table might best be described as a trace of the machine's expected operation.

Considerable further work would be required to turn the calculation described by the table into a set of cards the Analytical Engine could read to calculate all the Bernoulli numbers. As well as the operation cards themselves, the control deck would have to include special cards to tell the machine when to "back up" the deck to repeat a sequence of operations. Babbage had not settled on a format for these cards, or even on a mechanism for reading backward. A separate deck of "variable cards" specifying the numbers to be operated on would also be needed, as an operation card could be reused at different stages of a calculation to process different data.

Lovelace was part of her own time, not a visitor from the future or a lone superhero who invented programming and created the modern world. Through Babbage and her math tutor Augustus de Morgan, she was connected to contemporary thinking about mathematics, algebra in particular, and in the Notes she applied and extended those ideas in the context of the Analytical Engine, in detailed analyses and also in more general discussion. Lovelace was particularly interested in "cycles" of repeated operations, and the Bernoulli example involved two nested loops. In her Note E she reused "some of the notation of the integral calculus" to express loops symbolically, including ones with bounds set by computed values rather than constants.¹⁵ This notation enabled her to write a one-line expression she described as "represent[ing] the total operations for computing every [Bernoulli] number in succession"; see Figure 2.^f This expression provides the missing control structure and, with cross-reference to the table, it could be used to generate the deck of operation cards needed to run the full calculation. It is not a complete program for the Analytical Engine as an additional expression, not provided by Lovelace, would be required to define the structure of the deck of variable cards. In fact the idea of a program as we now understand it did not yet exist (an idea we will return to in a later column). It does, however, provide an interesting and neglected parallel to later efforts to develop abstract notions to specify computational processes.

Similarly, her suggestion that the Analytical Engine might be applied to the analysis and production of music was not a proposal for digital audio recording. One of the first topics she had discussed with de Morgan was the relationship between mathematics and music, an object of investigation since the time of Plato.¹⁷ She had observed the Analytical Engine was "an *embodying of the science of operations*, constructed with peculiar reference to abstract number as the subject of

f Lovelace's Note G.

Figure 2. Lovelace's one-line summary of the Bernoulli computation as a set of nested loops, using a novel mathematical notion of her own devising. The numbers 1–25 refer to the operations listed in the second column of her celebrated table.

(1...7), (24, 25), $\Sigma(+1)^{n}$ {(1...7), (8...12), $\Sigma(n+2)$ (13...23), (24, 25)} limits 1 to *n* limits 0 to (*n* + 2) those operations."^g Her musical proposal raised the prospect of embodying the abstract science in other areas, as George Boole was shortly to do in the case of logic. Her instincts, and her tragic personal story, were similarly emblematic of their time. She was Romantic with a capital "R," befitting her status as a daughter of Byron and echoing the era's belief in centrality of tortured genius and creative passion to greatness in both arts and natural science.

Grace Hopper

The second of Isaacson's "forgotten women" is Grace Hopper. Unlike Lovelace she was well known for her computing work during her lifetime, spending decades as a popular keynote speaker and appearing frequently on national television. Since her death in 1992 a guided missile destroyer and the Grace Hopper Celebration of Women in Computing (the leading annual event of its kind) have been named after her, as have a bridge and several buildings. She has been the subject of several full-length biographies and a documentary movie on her life is reportedly in progress.

Isaacson's treatment of Hopper in The Innovators is generally clear and accurate, lauding her for her collaboration with Aiken and for the pioneering work on compiler and programming language design she directed at Univac during the 1950s. However, a close reading hints at the ways in which inspirational superhero stories can hide the variety of contributions women have made to the history of computing. On page 117 Isaacson credits Hopper as "the technical lead in coordinating the creation of COBOL," a hugely important standard language for business programming that was defined jointly by several computer companies. Standards committee work is a difficult thing to dramatize, which is presumably why Isaacson gives CO-BOL no more than a passing mention, but as historian of technology Janet Abbate recently noted, his omission slights several women who, unlike Hopper, were on the technical committee in question. Among

g Lovelace's Note A, her italics.

them is Jean Sammett, who made the largest single contribution to the design of COBOL.^h Sammet has stated that Hopper was "not the mother, creator, or developer of COBOL," an idea Hopper reportedly endorsed by calling herself the "grandmother of COBOL" rather than its creator.¹⁰

Sammet's remarkable career began with work on programming languages for computer builders UNI-VAC, Sylvania, and IBM.³ She was the first woman to lead ACM, from 1974 to 1976, and was active in its special interest groups for decades. She pioneered the comparative and historical study of programming languages. Sammet has not been forgotten, as a fellow of the Computer History Museum and a winner of the IEEE Computer Pioneer Award, but sits on the wrong side of a growing chasm separating the handful of women chosen for public veneration from those whose names are known only to specialists.

This is an example of what sociologist of science Robert K. Merton called the "Matthew Effect," the tendency for the most famous person involved in any collaborative project, however peripherally, to be remembered by the public as the key contributor. He named it after a passage in Matthew's Gospel, noting that "unto every one that hath shall be given, and he shall have abundance: but from him that hath not shall be taken away even that which he hath." Over time the famous become more famous, while the moderately well known are forgotten.12 This is particularly true of those holding distinctions where the supply is deliberately limited, such as Merton's examples of the 40 "immortals" admitted to the French Academy or the winners of the Nobel Prize. It seems to us that a process of natural selection, in which a large pool of eligible pioneers compete for a handful of positions as female role models, is creating a similar dynamic.

The Women of ENIAC

Isaacson writes at length on "The Women of ENIAC," and excerpted this part of *The Innovators* in *Fortune* magazine as part of his launch cam-

paign. ENIAC, the Electronic Numerical Integrator and Computer, is remembered by historians as the first general-purpose electronic digital computer. It was planned and built at the University of Pennsylvania from 1943 to 1945, under contract to the United States Army, which justified its support for the rather bold foray into the technological unknown by pointing to an accumulating backlog of trajectory calculations needed to deliver accurate firing tables along with new artillery pieces. ENIAC missed the war, but when completed it tackled those calculations thousands of times faster than the women who had been working with desk calculators and was configured to tackle a range of other problems, including the first computerized Monte Carlo simulations and the first numerical weather calculations.ⁱ

ENIAC has always been the most famous of the early computers, but what it has been known for has changed over time. After finding fame as a scientific marvel and "giant electronic brain," it quickly became a yardstick against which the size, weight, and speed of newer computers could be flatteringly compared. In the early 1970s it was at the center of the longest trial ever held in Federal Court, during which a great deal of expensive expert and legal attention was focused on its claims to be the "first computer."

In recent years, however, it has been best known as a computer programmed by women. So here, too, the notion that Isaacson's characters were "forgotten" seems rather fanciful. Unlike Lovelace and Hopper, who were already well known in the 1980s, the six "Women of ENIAC" became famous more recently. However, Jean Bartik-the best known-has been inducted as a fellow of the Computer History Museum and has won the IEEE Computer Pioneer Award. Her obituary appeared in the New York Times. The women's work has been celebrated in countless blogs, books,

h http://bit.ly/1go319Y

i Material on ENIAC here is sourced from our forthcoming book T. Haigh, M. Priestley, and C. Rope, *ENIAC in Action: Making and Remaking the Modern Computer.* MIT Press, Cambridge, MA, 2016.

and scholarly articles and in two documentary films.

Isaacson's retelling of their story is sourced primarily from Bartik's engaging memoir, Pioneer Programmer and from a number of oral history interviews.² He observed that while "the boys with their toys thought that assembling the hardware was the most important task, and thus a man's job" in fact "all the programmers who created the first generalpurpose computer were women." This odd verbal flourish suggests that Isaacson, despite his love of innovation, does not have a very clear idea of what creating a computer involves. Programmers, however gifted, do not usually design computers and design work was already finished when the women in question were chosen for ENIAC work in the summer of 1945. We are struck that Isaacson rhetorically downgrades the contribution of the designers and engineers to "assembling the hardware." Elsewhere he spins a statement from J. Presper Eckert, the project's lead engineer, that he double-checked the calculations of his subordinates into an assertion that Eckert would "hover over the other engineers and tell them where to solder a joint or twist a wire." He seems to have no sense of computer engineering work as something different from tinkering with a soldering iron.

One startling fact was overlooked by Isaacson and almost everyone else who has written about ENIAC. The cliché that the machine was assembled by men but programmed by women is just not true. Dozens of women, employed as "wiremen," "assemblers," and "technicians" worked shifts through 1944 and 1945 to build ENIAC, threading miles of wire through the machine and soldering approximately half a million joints. Other women drew blueprints and helped with administration. While Bartik and her colleagues were given belated recognition for their work on ENIAC, more than 50 women worked on the machine in 1944 alone. Their names are preserved only in the project's accounting records and personnel forms. They are truly forgotten.

We also believe the word "programming" does not do a good job of capturing the actual work of Bartik and her colleagues. When programming developed as a distinct job, in the mid-1950s, it was generally separated from the work of operating the computer or any ancillary punched card units. Programmers sat with pencils and coding pads, writing instructions. These were punched onto cards by keypunch women, the data entry clerks of their day. Operators tended to the machine itself, preparing the computer and its peripherals for a job, loading the appropriate card and tape media, and extracting printed output.

The women of ENIAC combined elements of all three roles, but operations work was what they were hired to do and it occupied most of their time with the computer. They were chosen for their skill as human computers, carrying out manually the tasks ENIAC would process electronically. Their initial training was in punched card machine operation. ENIAC's tiny writable electronic memory (just 200 decimal digits) meant running a reasonably ambitious job involved interleaving computer steps, in which decks of cards were run though ENIAC for automatic computations, with manual steps in which the cards were punched with new values, sorted, collated, or run through the tabulator for printing.

The initial cohort of six operators had to understand how ENIAC worked, so they could set it up for a new job,

Lovelace was part of her own time, not a visitor from the future or lone superhero who invented programming and created the modern world. which involved many hours of plugging cables and setting switches, and monitor its progress. Because of this expertise they also played an important preparatory role in many of the applications run on ENIAC, partnering with scientific and mathematical users to figure out the "set-up" needed to configure the machine to carry out the appropriate series of mathematical operations. These set-ups implemented loops, subroutines, and the other elements we now associate with programs (although they were recorded visually, as a schematic of the switch and wiring settings needed on ENIAC's various units, rather than as textual instruction codes).

It is this preparatory work that has caused them to be remembered as programmers, though their contributions here are frequently exaggerated. For example, it is often claimed ENIAC programming was an afterthought that received no attention from the engineering team. The women's realization, when planning trajectory computations, that ENIAC's "master programmer" unit could be used to implement loops is sometimes taken as a breakthrough moment in programming history.

In fact, ENIAC's designers had paid considerable attention to the trajectory calculations early in the design process, culminating at the end of 1943 in the creation by Arthur Burks of elaborate set-up and timing diagrams for this problem. This work profoundly shaped ENIAC's final design, particularly its master programmer, and introduced the diagramming and planning techniques later used by Bartik and her colleagues. The delight the women expressed in later interviews in grasping techniques such as looping are familiar to later generations of computing students but do not indicate they had discovered an unanticipated use for the master programmer.

Many of the errors and omissions in Isaacson's take on this history come from the oral histories and existing books he relies on, and reflect a conventional wisdom that has developed over time. His most novel suggestion, apparently based on a misreading of Bartik's memoir, is that she contributed to discussions on computer design held between John von Neumann and the ENIAC project engineers held at the Moore School in early 1945.^j By April these led to the *First Draft of a Report on the EDVAC*, the first statement of the crucial ideas of modern computer architecture. While the proper balance of credit for these ideas has been much disputed, Isaacson is unique in assigning Bartik, who had not yet been recruited for ENIAC work, a role in formulating the new instruction set. This error will surely be repeated by others, further building the superhero myth.

Our intent is not to belittle the very real contributions of the ENIAC operators to computing history, but to argue that forcing them into the role of the "first programmers," as is commonly done, hides most of the reality of their actual work. Operating ENIAC, and its ancillary punched card machines, was highly skilled work that was absolutely vital to the machine's success. To this day programming accounts for a fairly small proportion of overall IT work, and the demand for system administrators, network technicians, facilities management staff, and other hands-on IT practitioners is greater than ever. By the 1960s, however, computer operations work was seen as a lower paid and lower skilled occupation-blue-collar shift work against the white-collar work of programming. Perhaps celebrating hands-on operations work would mislead impressionable young women. As Wendy Hui Kyong Chun has noted, "reclaiming these women as the first programmers...glosses over the hierarchies...among operators, coders, and analysts."4 Remembering the ENIAC operators only as programmers implies gender matters to history but social class does not.

Conclusion

A great deal is at stake here. Women are conspicuous by their absence in computer science classrooms and in the programming teams, data centers, and computing research labs of America.⁶ Computer technology has been actively claimed by some men as their exclusive domain, as in the recent "Gamergate" campaign to deter women from writing about video games. While most areas of science and engineering are gradually becoming more balanced in their gender representation, computer science has slipped backward over the past 30 years. This has prompted a great deal of hand wringing and countless blue ribbon committees, research reports, proposed interventions, and volunteer campaigns to recruit and retain more women in various areas. ACM has been heavily involved in this area. Maria Klawe, ACM president from 2002–2004, spearheaded work at Princeton and, more recently, as President of Harvey Mudd College. ACM-W, which "supports, celebrates, and advocates internationally for the full engagement of women in all aspects of the computing field" is open to all members without additional cost. In 2006, ACM ended a 40-year streak for the boys' team when it presented a Turing Award to Frances Allen. Women won again in 2008 and 2012.

History is playing an increasingly visible role in this struggle. One widely held concern is that girls are not exposed to images of female programmers or information technology professionals. Films about Jobs, Gates, Turing, and Zuckerberg encapsulate the broader tendency of mass media to present computers primarily as an object of fascination for brilliant but spectacularly awkward men. In their own lives teenage girls are less likely than boys to find peers with an interest in programming or computer technology. They have little reason to assume that software development is work they would find interesting or be good at.

The past has therefore been mined in search of women who might provide girls with inspiring counterexamples. The goal of convincing women to enter computing might sometimes seem at odds with the responsibility of historians to provide accurate and nuanced stories. In such a battle the force of numbers would not be with the historians, who find intrinsic value in the past of a field that is relentlessly focused on the future. In contrast, most peo-

We run the risk of hiding a rich historical tapestry, full of thousands of little figures carrying out many vital tasks.

ple with a connection to computing encounter history only occasionally as disconnected fragments, for the time it takes to read a blog post, watch a movie, or walk through a science museum exhibit.

We believe good history is not just important to specialists, and that it will ultimately prove more inspiring and more relevant than superhero stories. Isaacson's book has some distinctive strengths and weaknesses, but we have used it here primarily as an accessible, and influential, summary of the emerging public understanding of the contribution of these superhero coders to the invention of the modern world. This has many parallels with the recent film, The Imitation Game, in that both are designed to flatter their audiences by making them feel superior to the sexist, homophobic, or otherwise flawed characters who conspire unsuccessfully to prevent their heroes from reaching their special destinies. Stories of this kind inflate their own importance, by pretending a certain idea, person, or invention has direct and singular responsibility for some hugely complex chain of later events that leads to our modern "digital world." Their authors are fixated on firsts and origins, and glorify flashes of abstract insight over slow and incremental technological change driven by the needs of users.

The conception of history as the deeds of a few "great men" has problems that are too fundamental to correct simply by elevating a few "great women" to the pantheon. Indeed, this strategy devalues many of the contributions made by women. Nobody has

j Isaacson's discussion, on pages 106–107, is apparently based on pages 118–119 of Bartik's memoir. She is discussing work with von Neumann in 1947 to plan a new instruction set for ENIAC, rather than 1945 design work on EDVAC.

chosen to celebrate "The Key Punch Operators Who Invented the Digital Universe" or "The Lone Computer Assembler Who Built the Modern Computer." The obsession with discovering the first coders, intended to empower women, has obscured the importance of other kinds of computer-related labor that was carried out by women for decades. The quest for "girls who code" is erasing the history of "women who operate." One of the reasons such work was seen as low status and unimportant is that it was carried out by women, and any attempt to challenge historical sexism should dispute that assumption rather than tacitly endorsing it. Scholars have recently looked more systematically at the historical participation of women in computing, most notably in Janet Abbate's Recoding Gender,¹ which followed the history of women in computing from the 1940s to the 1970s, and in the collection Gender Codes edited by Thomas Misa.¹³ Nathan Ensmenger has looked at rhetoric around the gendering of programming in the 1960s and 1970s.7 Marie Hicks and Ian Martin have both explored the work of computer operators, looking at the intersection of class and gender during the same era.9,11 Laine Nooney has been exploring the career of Roberta Williams, cofounder of Sierra On-Line and designer of the King's Quest series of adventure games, as a different model for female success.¹⁴

Stories about superheroes make great entertainment, as testified to by the huge success of the Avengers films and the other movies and licensed television series based on the Marvel characters. The superhero narrative is not, however, the best way to understand history. Isaacson's celebration of Ada Lovelace, Grace Hopper, and the "Women of ENIAC" as the creators of computer programming is part of a much broader movement to make these women into recruiting sergeants for computer science. The colorful feats sometimes attributed to these women can overshadow their actual historical contributions, which were rather more nuanced and rather less dependent on superpowers. In particular, the concept of "programming" has been awkwardly projected back in time, in a way that misrepresents their accomplishments and squanders the chance to celebrate a broad range of computer-related occupations such as mathematical analysis, operations work, and management.

More generally, we run the risk of hiding a complex historical tapestry, full of thousands of little figures carrying out many vital tasks, behind a gaudy poster depicting a small team of superheroes. The quest for "genius" devalues the vital contributions of millions who are merely creative, intelligent, hard-working, and lucky enough to be in the right place at the right time. Superhero stories have little time for ordinary humans, who exist only to be endangered or rescued. Reducing the story of women in computing to the heroics of a handful of magical individuals draws attention away from real human experience and, counterproductively, suggests that only those with superhuman abilities need apply.

References

- Abbate, J. Recoding Gender: Women's Changing Participation in Computing. MIT Press, Cambridge, MA, 2012.
- Bartik, J.J. Pioneer Programmer: Jean Jennings Bartik and the Computer that Changed the World. Truman State University Press, Kirksville, MO, 2013.
- Bergin, T.J. Jean Sammet: Programming language contributor and historian, and ACM president. *IEEE Annals of the History of Computing 31*, 1 (Jan.–Mar. 2009), 76–85.
- 4. Chun, W.H.K. Programmed Visions: Software and Memory. MIT Press, Cambridge, MA, 2011, 34.
- Cohen, I.B. Howard Aiken: Portrait of a Computer Pioneer. MIT Press, Cambridge, MA, 1999, 61–72.
- Cohoon, J.M. and Aspray, W., Eds. Women and
- Information Technology: Research on Underrepresentation. MIT Press, Cambridge, MA, 2006.
- Ensmenger, N. The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise. MIT Press, Cambridge, MA, 2010.
- Essinger, J. A Female Genius: How Ada Lovelace, Lord Byron's Daughter, Started the Computer Age. Gibson Square, London, 2014, 174.
- Hicks, M. Only the clothes changed: Women computer operators in British computing and advertising, 1950–1970. *IEEE Annals of the History of Computing*, 32, 4 (Oct.–Dec. 2010), 5–17.
- Lohr, S. Go To: The Story of the Math Majors, Bridge Players, Engineers, Chess Wizards, Maverick Scientists and Iconoclasts—The Programmers Who Created the Software Revolution. Basic Books, New York, 2001, 52.
- Martin, I. Structuring information work: Ferranti and Martins Bank, 1952–1968. *Information & Culture 47*, 3 (2012), 312–339.
- 12. Merton, R.K. The Matthew effect in science. *Science 159*, 3810 (1968), 56–63.
- 13. Misa, T.J. *Gender Codes*. IEEE Computer Society Press, Hoboken, NJ, 2010.
- Nooney, L. A pedestal, a table, a love letter: Archaeologies of gender in videogame history. *Game Studies* 13, 2 (2013).
- Priestley, M. A Science of Operations: Machines, Logic and the Invention of Programming. Springer, 2011, 41–44.
- 16. Stein, D.K. Ada: A Life and a Legacy. MIT Press, 1985, 92.
- Toole, B.A., Ed. Ada, the Enchantress of Numbers: A Selection from the Letters of Lord Byron's Daughter and Her Description of the First Computer. Strawberry Press, Sausalito, CA, 1992, 245.

Further Reading

Abbate, Janet.

Recoding Gender: Women's Changing Participation in Computing. MIT Press, Cambridge, MA, 2012. This starts with ENIAC and Colossus and moves forward as far as the 1970s. As a small book on a huge topic it focuses on particular examples, but these are well chosen and give a sense for broader changes in the IT workplace.

Bartik, Jean Jennings.

Pioneer Programmer: Jean Jennings Bartik and the Computer that Changed the World. Truman State University Press, Kirksville, MO, 2013. Bartik's memoir gives a lively and idiosyncratic view of the ENIAC story.

Beyer, Kurt W.

Grace Hopper and the Invention of the Information Age. MIT Press, Cambridge, MA, 2009. The most detailed account to date of Hopper's work with Aiken at Harvard and at Univac, putting her achievements into the broader context of early computing.

Fritz, W. Barkley.

The Women of ENIAC. *IEEE Annals of the History of Computing 18*, 3 (Fall 1996), 13–28. Full of material Fritz elicited from the women themselves, this article played a crucial role in bringing their contributions to broader awareness.

Padua, Sydney.

The Thrilling Adventures of Lovelace and Babbage: The (Mostly) True Story of the First Computer. Pantheon, 2015. This charming, funny, and heavily footnoted graphic novel turns Babbage and Lovelace into literal superheroes. It is no less accurate than many other popular retellings of the story but has the good grace to admit that much of its action takes place in a parallel universe.

Stein, Dorothy.

Ada: A Life and Legacy. MIT Press, Cambridge, MA, 1985. Published 30 years ago, but still the most reliable biography of Ada Lovelace.

Swade, Doron.

The Difference Engine: Charles Babbage and the Quest to Build the First Computer. Viking Penguin, New York, 2001. This wonderfully engaging book has two halves. One describes Babbage's own computing career; the other the ultimately successful project led by Swade to complete his plans to build the "difference engine," the earlier and less complicated of Babbage's two planned computers.

Thomas Haigh (thaigh@computer.org) is an associate professor of information studies at the University of Wisconsin, Milwaukee, and immediate past chair of the SIGCIS group for historians of computing.

Mark Priestley (m.priestley@gmail.com) is an independent researcher into the history and philosophy of computing.

Copyright held by authors.