

Incorporating a Theme Through Literature

Dr. Robin K. Hill, University of Wyoming

Dr. Hill is an adjunct professor in both the Wyoming Institute for Humanities Research and the Philosophy Department of the University of Wyoming, and a Lecturer in Computer Science. She currently writes a blog on the philosophy of computer science for the online Communications of the ACM. Her teaching experience includes logic, computer science, and information systems courses for the University of Wyoming, University of Maryland University College (European Division), State University of New York at Binghamton, Metropolitan State College, and others. She holds a Ph.D. in Computer Science, an M.S. in Management Information Systems, an M.A. in Mathematical Logic, and a B.A. in Philosophy.

Incorporating a Computing Theme through Literature

May 15, 2021

Abstract

Even in a first-year seminar in a technical field like computer science, the assignment of a classic work of literature touching on a theme related to computing can provide rich material for demonstrations, exercises, and thoughtful analysis on the relations between society and technology.

Introduction

The First-Year Seminar in Computer Science offered by my department is taken by a variety of students bringing a variety of backgrounds and exhibiting a variety of aptitudes and interests. Myriad challenges accompany the goal of turning out college students well prepared to continue their university education. A classic novel, chosen for a computing-related theme, provides not only a rich literary experience but also practice in concentrated reading and a platform for various exercises on aspects of computing. As far as this author can tell, scholarly work on the incorporation of literature into computing courses is lacking. Endeavors to integrate humanities into computing (and other technology) fall into categories such as understanding technology through science fiction [1], design for fruitful user experience [2], ethics and scrutiny on social effects [3], and communication and collaboration skills [4]. This initiative, a straightforward reading assignment, reaching outside the obvious subject matter, fits into none of those categories. This paper suggests the benefits that it can confer in an introductory computing course.

The Scene

The Computer Science First-Year Seminar (FYS) is one of about 60 such seminars across campus. The university studies (general education) program requires incoming freshmen and other first-year students to take one from this selection of seminars on different subjects, not necessarily in their major field if they are among the few who have chosen a major already. The underlying FYS goal, teaching these high school graduates to be college students, is rendered formidable by not only the diversity of backgrounds but the fact that our campus is open to virtually all high school graduates in the state. For some, the protocols and practices of college are foreign and mysterious [5].

The Course

The Computer Science section attracts the range of students already described, some with curiosity but no experience in the field, and some with no overt interest. The challenge is to introduce enough technical material to construct a solid, though sparse, conceptual framework for computation, while also stimulating student interests and introducing concerns of greater scope. Although the computing material is not the subject of this paper, the reader should understand that about half of the course covers computing and programming, with the other half based roughly on the subject matter commonly called “computers and society.” Mostly for that latter subject, we use “Blown to Bits” by Abelson, Ledeen, and Lewis as a nominal textbook [6]. It’s free and online, criteria which also drive the choice of classic novel. Herein, we explain how that novel affords lessons related to both halves of the content; a complete account of the material covered in the course is beyond the scope of this work.

Technical and Programming Content

A few words about the technical content will suffice. Predictably enough, the class performs rudimentary programming (in Java with BlueJ). The technical content covers the barest fundamentals that allow a grasp of the computational paradigm, from bits, bytes, and their diverse interpretations to variables and types to the control structures of algorithms. At the start, Java program skeletons are assigned, to be completed with minimal code added at critical spots; later, students study full programs and outline algorithms. Balancing the needs of the video gamer, the app hacker, and the baffled youth with no programming experience poses a challenge that this author has not met with stunning success. We can only hope and trust that those who fume with impatience at the “toy” coding exercises become intrigued with other parts of the course. A pedagogical goal for this half of the course is sufficient grasp of programming to understand that a computer virus is a set of instructions that makes its way into the CPU, and the mechanisms that allow damage to occur.

Computers and Society Content

The FYS in computer science investigates and discusses current issues that have emerged in public discourse, such as (in the spring of 2021), privacy of personal data, information disorder, cyberabuse, and issues of artificial intelligence. And viruses—we investigate and discuss biological viruses, the integrating (computing-related) theme for this term. That choice was motivated, of course, by the COVID-19 coronavirus, a ready appeal to current events. A pedagogical goal for this term is sufficient grasp of biological viruses (actually, pathogens in general) to probe the analogy to the damage done by computer viruses; other terms may offer other themes with pedagogical goals appropriate to them.

Assigning Literature

This course’s selection of a theme connects the humanities to computer science, with perhaps a degree of contrivance. In past years, we read “Frankenstein” under the theme of the *unintended consequences of technology*, and “The Scarlet Letter” under the theme of *public shaming*. During the spring semester of 2021, the classic work is “Castle Richmond” by Anthony Trollope [7], set

during the Irish Potato Famine, and the theme is *the damage done by viruses*. (The author is fully aware, after exacting instruction by associates in the real sciences, that the potato blight is not a virus, but rather the micro-organism *Phytophthora infestans*, an oomycete, and so informed the class.)

Justification

The class hears reasons for reading in general, independent of subject matter, as follows: You must learn to read in a sustained way, comfortably, for long periods of time. You must put aside distractions and pay attention. You must allow yourself to get involved. Reading a book, or an article, or a document, from start to finish is good habit that will pay off in classes now and in jobs later. A novel provides good practice.

Because the novel contains XLIV chapters, students are not strictly required to read the whole book, a grudging concession to reality. (They take turns giving short chapter summaries, under strong encouragement to read the whole book.) The Roman numerals, by the way, help to introduce discussion of number systems. The class also reads program source code, a type of close reading different from that applied to literature, but alike in the concentration required by each that can presumably be strengthened by the other (although no research was found that addresses this).

Many contemporary commentators lament the erosion of humanities enrollments in favor of other college subjects [8, 9]. Certainly the liberal arts need promotion not just because they constitute nice things in society, but because they help us query conditions and solve problems. The motive here is not an assault on curriculum committees, but a broadening and refinement of the student experience in higher education, most naturally done through humanities. The text provides particular affordances as follows.

Text Exercises

A lengthy text provides a platform for some introductory research exercises—finding an unknown word, counting occurrences, and defining it for the class; writing thesis statements for given passages; and summarizing chapters, in written and spoken form. These exercises can then be repurposed for more technical work: Words can be turned into character strings, and graphics can be practiced with story scenes. We discuss algorithms for pattern-matching that underly text search. See below for some of the connections from the text exercises to other subjects.

Text and Computing Exercises

- Word Definition (from novel)
- Text search (for words in novel)
- Fact searches that are difficult for text, e.g., ages of characters (in novel)
- Roman numerals
- Representing characters with standard codes (words from novel)
- Graphics—Icon design related to the story of the novel

- Steganography—encoding message from novel
- Web page—Simple web page (in plain HTML) illustrating a theme from the novel, using the icon

Let it be understood that the course covers all the basics such as bit strings and byte interpretation, beginning architecture and CPU cycle, simple programming, and so forth; this list just shows the direct pedagogical uses of the classic novel for basic computing concepts.

The Human Condition

The reading can also be probed for human aspects of narrative situations and comparisons to their manifestations in modern technology; these analogies should be handled carefully to deter contrived or artificial connections.

In the term of “The Scarlet Letter,” we compared Hester Prynne’s circumstances to those of Justine Sacco, shamed and fired for tweeting a joke that could be interpreted as a racist insult, as a weak attempt at humor, or even as amusing, and asked whether the Internet imposes suitable consequences for such actions. In the term of “Frankenstein,” we compared the damage done by the monster to other unintended consequences of technology such as the hostility that develops on the Internet, and asked whether features of the Internet foster bad conduct. In the current term, of “Castle Richmond,” we use the theme of the damage done by viruses to prompt and synthesize reflections on these matters:

1. The exploitation by a malicious attacker of a system’s own mechanisms;
2. The heavy dependence of a society on a monoculture, whether in food or technology (hardware, software, social media);
3. Detection and treatment of biological and computer viruses, mutation, and the nature of immunity in disparate realms;
4. What goes wrong when conscientious people take on a social problem—particularly pointed in Castle Richmond’s depiction of famine relief, and also visible in well-intended computing initiatives such as One Laptop Per Child [10];
5. The pressure of learning how to alleviate a disaster during the disaster itself.

Such issues can be addressed in short essays, class discussions, and many other pedagogical approaches, contributing to social and civic education. Research paper and group presentation assignments lead students to inquire into the social context of technology through the Internet setting and its attendant issues. A more abstract goal is the integration of learning on disparate subjects, that is, a demonstration to a first-year student that course material can lead to new subjects of interest in their own right, that these subjects can help the student to achieve, flourish, and contribute, and that college can enhance many aspects of life.

Summary

Integration of Computer Science and the Humanities

In this introduction to computer science, the classic novel is a means to ends as well as an end in itself. As a shameless humanist in technology (a B.A. in philosophy and Ph.D. in computer science), I want the students to think at a high level about the benefits, issues, affordances, and limitations of computing, and of all technology. The novel provides connections between the two parts of the course, the technical and the social, and the theme that crosses from one to the other, as a bridge between the computer virus material and the biological virus material, models the very idea of integration across diverse studies. A capstone question, how facilities of the modern Internet might help save people in a disaster brought about by a plant or animal virus, elicits yet another perspective on the relationships among these subjects.

Results

The author regrets that no quantified results are available, no controlled experiments having been undertaken. Students have not been systematically queried for their opinions; the author retains some skepticism that first-year students can assess the success of the pedagogy to which they are subject. My experience in teaching computer science (40+ years) suggests a weak achievement of success in that students do not seem to be put off by the novel, and seem to be intrigued by the associations revealed to them by the exercises that extend through computing, literary, and social realms. The coronavirus precautions under which we have been operating discourage engaged discussion and drive this teacher to an unfortunate increase in lecture and cold-calling on students by name from time to time for answers to prompts, with mixed results. The author has to admit that even if students said that they did not like it, she would probably continue with this practice in hopes of justification by their future enlightenment and appreciation.

Conclusion

We hope that the First-year Seminar teaches our incoming students *how* (and also *why*) to go to college, cultivating higher-order thinking skills and inspiring students to explore the world of ideas. We further hope that they learn some computer science; this instructor is happy if they come to understand different interpretations of bit strings and how algorithmic constructs work. Because many exercises in introductory programming and applications of computing call for some kind of text, a literary work serves purposes in that realm. The incorporation of the novel is also intended to foster appreciation for non-technical studies as well as the interpretive skills that serve those studies, and this instructor would be quite pleased to cultivate an appreciation for literature as an end in itself.

References

- [1] E. Burton, J. Goldsmith, and N. Mattei, "How to teach computer ethics through science fiction," *Commun. ACM*, vol. 61, no. 8, p. 54–64, Jul. 2018. [Online]. Available: <https://doi.org/10.1145/3154485>

- [2] J. Schummer, B. MacLennan, and N. Taylor, "Aesthetic values in technology and engineering design," in *Philosophy of Technology and Engineering Sciences*, A. Meijers, Ed. North Holland, 2009, section 4; author Bruce MacLennan.
- [3] S. Zilliox, J. Smith, and C. Mitcham, "Teaching the ethics of science and engineering through humanities and social science," *Teaching Ethics*, vol. 16, no. 2, pp. 161–183, 2016.
- [4] V. M. Arms, "Personal and professional enrichment: Humanities in the engineering curriculum," *Journal of Engineering Education*, vol. 82, no. 2, pp. 141–146, 1993.
- [5] K. McKay. (2021) High school vs university. From the Tommorrow's Professor mailing list of Rick Reis, Post #1857 at <https://tomprof.stanford.edu/>. [Online]. Available: <https://mansci045.uwaterloo.ca/pageHS.html>
- [6] H. Abelson, K. Ledeen, and H. Lewis, *Blown to Bits: Your Life, Liberty, and Happiness after the Digital Explosion*, 1st ed. Addison-Wesley, 2008, freely available at <http://www.bitsbook.com>.
- [7] A. Trollope, *Castle Richmond*. Dover Pulications, Inc., 1984, republication of 1860 edition.
- [8] L. Shamir, "A case against the STEM rush," *Inside Higher Ed*, 2020. [Online]. Available: <https://www.insidehighered.com/views/2020/02/03/computer-scientist-urges-more-support-humanities-opinion>
- [9] K. Leetaru, "Why computer science needs the humanities," *Forbes*, 2019. [Online]. Available: <https://www.forbes.com/sites/kalevleetaru/2019/08/06/why-computer-science-needs-the-humanities/?sh=11641d97f45f>
- [10] A. Robertson, "Olpc's \$100 laptop was going to change the world — then it all went wrong," *The Verge*, 2018. [Online]. Available: <https://www.theverge.com/2018/4/16/17233946/olpcs-100-laptop-education-where-is-it-now>