Report of the SIGCSE Committee on Computing Education in Liberal Arts Colleges

Doug Baldwin
Dept. of Mathematics
SUNY Geneseo
Geneseo, NY, USA
baldwin@geneseo.edu

Amanda Holland-Minkley
Department of Computing & Information Studies
Washington & Jefferson College
Washington, PA, USA
ahollandminkley@washjeff.edu

Grant Braught
Dept. of Mathematics and Computer Science
Dickinson College
Carlisle, PA, USA
braught@dickinson.edu

The 2016 creation of the SIGCSE Committee on Computing Education in Liberal Arts Colleges gave voice to a large population teaching computing as a component of liberal education. For the past two years the committee has worked with this population and the larger SIGCSE community, and reviewed the literature on liberal arts computing, to identify the needs of computing educators at liberal arts colleges. The committee found an overwhelming desire for an organization to support and represent those educators, and recommends that work begin immediately to create such an organization.

Introduction

Questions concerning the role of computing programs in liberal arts colleges have historically been fraught. In the early days of computing education, many educators struggled to make the case that computer science (let alone more applied computing fields) belonged in liberal arts colleges at all [6,12]. Even where computing was accepted as appropriate, contemporary curriculum recommendations failed to reflect the priorities of the liberal arts [6,14]. Today most, although not all, liberal arts colleges offer computing programs, and such colleges are respected sites for computing education, as illustrated by the many liberal arts colleges among the course and curriculum exemplars in the CS2013 curriculum guidelines [15]. Nonetheless, a sense that the liberal arts and computing somehow make an uneasy pair still lingers.

The SIGCSE Committee on Computing Education in Liberal Arts Colleges was created in 2016 to examine this "uneasy pair." The committee's charge focused on two complementary questions:

- Does the liberal arts computing education community need some sort of organization to support its activities?

- Does the larger computing education community need some identified organization it can turn to as the voice of liberal arts computing?

This report presents the committee's findings. It describes the process by which the committee carried out its work, the definition of "liberal arts" that framed that work, and the nature of liberal arts computing programs today. With this background, the report then turns to the main findings regarding the focal questions, and concludes with recommendations for future action.

Process

The committee held a face-to-face organizing meeting at SIGCSE 2016. Business since then has been conducted via the committee's mailing list (sigcse-libarts-comm@acm.org), progress report special sessions at SIGCSE 2017 and 2018 [2,3], and related birds-of-a-feather sessions.

As of June 2018 the committee mailing list had 111 subscribers, representing at least 77 distinct colleges and universities in the United States (as determined by email addresses; apart from a handful of "acm.org," "gmail.com," and similar addresses, all are in US ".edu" domains). The committee thus has a good representation of US higher education, but little if any non-US or K-12. This is perhaps not surprising, given that liberal arts colleges are seen as a predominantly US form of higher education.

The committee drew on a number of sources to inform its deliberations, including…

- A survey of committee members' perceptions and concerns conducted in the summer of 2016,
- A survey of all SIGCSE members concerning their interactions with or for liberal arts computing conducted in the spring of 2018,
- Discussions at a special session [3] and birds-of-a-feather meetings at SIGCSE 2017,
- Discussion during a special session at SIGCSE 2018 [2],
- Discussion on the SIGCSE-LIBARTS-COMM mailing list, and
- Evidence gathered from the computing education literature.

These sources range from purely anecdotal (conference and mailing list discussions) to modestly rigorous (survey results) to refereed publications (existing literature). Of the two surveys, the first was directed only at committee members. There were 38 responses from 94 mailing list subscribers at the time, for a 40% response rate. The second survey was sent to the sigcse-members mailing list, and received 63 responses, a negligible response rate. Committee members are obviously self-selected for interest in liberal arts computing, and of the 63 responses to the second survey, 58 described themselves as "actively involved in liberal arts computing education." So the survey results reflect only the views of the liberal arts computing education community, and indeed that subset of the community that is passionate enough to be involved in the committee and SIGCSE and willing to fill out surveys. Nonetheless, we believe that the committee's large size and the attempt to support findings with other literature mean that our results represent that liberal arts community well.

What is “The Liberal Arts”?

The committee’s first order of business was to agree on what we understood by “liberal arts.” To many people, the phrase means a set of subjects centered on the arts and humanities, and the phrase “liberal arts college” means a small undergraduate institution with a curriculum built around these subjects. We, however, found that more nuanced definitions better fit modern higher education. For example, the essays in [10] explore the liberal arts and liberal arts colleges in the mid-twentieth-century United States as sources of career education rooted in a rich political, historical, and social context, identifying particularly their function in “seeing to it that the technical subjects which are now socially necessary acquire a humane direction.” Roth [22] sets these views in a historical context stretching from Benjamin Franklin and Thomas Jefferson to modern thinkers, a context that sees higher education leading students into careers, preparing them for engagement in democratic government, empowering and liberating them personally, and advancing society collectively. The Association of American Colleges and Universities uses the phrase “liberal education” to identify such an education that serves career, civic, and societal needs throughout students’ lives [1].

The committee followed these broad definitions of liberal arts as liberal education. Thus, we understand our title’s reference to “computing education in liberal arts colleges” as referring to institutions that pursue a philosophy of higher education that emphasizes preparing students for the full range of thinking they will face throughout their lives: thinking in the service of a career, thinking in order to participate in civic affairs and society generally, thinking in order to have a fulfilling personal life, etc.

Our view of the liberal arts has two important consequences for our work. First, the computing disciplines, for all that they may be “technical” fields and for all the emphasis they may place on preparing computing professionals, definitely have a place in liberal arts colleges. Second, liberal (arts) education can and does happen in many institutional settings—certainly in the stereotypical small undergraduate colleges, but also in the undergraduate programs of many research universities, embedded in the curricula of some professional schools, etc. Elements of a liberal education can and do appear at schools that do not necessarily think of themselves as fundamentally liberal arts colleges. The extent to which individual students, faculty, or programs experience and embrace liberal education’s values varies within institutions, even institutions that fit the traditional liberal arts model. “Liberal arts computing education” thus covers any computing courses or curricula designed with the goal of educating students broadly for life in addition to any professional or career goals that are also supported. We hope that this committee’s findings will be useful to all people or institutions who are interested in computing as a component of such an education, regardless of how they label themselves.

Characteristics of Liberal Arts Computing Programs

Our surveys, discussions, and literature review found a number of distinctive features of liberal arts computing programs. We classify these features as programmatic ones distinguishing liberal arts computing programs from

other computing programs, features relating computing to the larger liberal arts mission, and consequences of liberal arts programs often being located in small colleges.

Program Size and Structure

The canonical liberal arts computing program consumes a relatively small fraction of students' total instructional hours, focusing on central principles rather than specific products or practices [17]. Reality, however, is more complicated than this single canonical view. The median program reported by committee members in our 2016 survey is indeed fairly small, accounting for about ⅓ (38%) of the total requirements for graduation from the institution (measured in credit hours, courses, or other units an institution may use—we give program sizes as fractions of the graduation requirement to allow comparisons independent of local units). However, there is considerable variation from this median, with the smallest program accounting for 29% of the graduation requirement and the largest 63%. This size distribution matches the five case studies in [4] well, where the programs range from 31% to 60% of graduation requirements, with a median of 43%. Roughly half of the committee's programs lead to a Bachelor of Arts (BA) degree, while half lead to a Bachelor of Science (BS), but only 4 of the 38 programs (10%) are ABET[1] accredited. Computing programs are offered under a wide variety of names and contexts, ranging from "computer science" housed in its own department to computer science concentrations within a mathematics major to a "media arts and sciences" major.

Streib and White's survey of program sizes at 15 Illinois liberal arts colleges [24] found programs ranging from 24% to 32% of total graduation requirements (assuming 120 semester hours required for graduation), at the low end of the range we found. This may reflect changes in computing curricula, and liberal arts colleges responding to pressure to increase the size of their computer science programs, in the years since Streib and White's survey. Two new ACM model curricula have been published since that survey, each with a larger recommended core than the 2001 model curriculum [15]. Also suggesting that liberal arts computing programs are responding to larger trends, Streib and White found no departments with ABET accreditation where the committee found four.

We compared the 30 liberal arts computer science programs in our 2016 survey to the top 25 computer science programs at national universities as identified by US News [25]. One of the clearest differences is in the type of degree granted, as seen in Figure 1. The liberal arts programs are about evenly split between BA and BS degrees; some programs grant either with slight differences in requirements, and one grants a Bachelor of Computer Science (BCS) degree. The nationally ranked universities, on the other hand, have a significant number of programs leading to a BS in Engineering (BS Eng) degree, as well as programs leading to a standard BA or BS.

[1] The main accreditor of post-secondary engineering and technology programs in the United States, also active in many other countries.

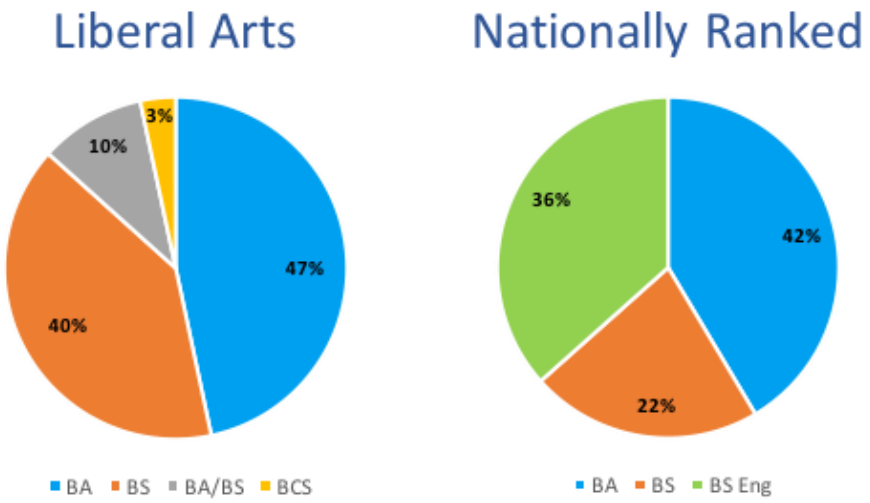

Figure 1. Comparison of computer science degree types between liberal arts institutions and nationally ranked universities.

Structurally, the liberal arts computer science programs differed from computing programs at the nationally-ranked universities in expected ways. Across both institution types, BA programs were on average smaller as a fraction of students' overall undergraduate coursework (35-40%) than were BS programs (45-53%) which in turn were also smaller than those that identified with engineering or were ABET accredited (59-70%). By the same measure BA, BS and ABET programs that identified as liberal arts were smaller (36%, 45%, 59% respectively) than the corresponding degree programs at national universities (40%, 54%, 71%). By and large, the difference in size tended to translate into fewer electives in the liberal arts programs—only 21% of the credits in their majors were electives versus 26% for the nationally ranked programs. The non-computing requirements in the programs were similar across liberal arts and national university programs and included such things as additional science courses, technical writing, communications and ethics. National university programs were more likely to require coursework in specific sciences (physics, chemistry, biology). ABET accreditation was much more common among nationally ranked programs than among liberal arts programs (25% of the programs vs 10%). Figure 2 summarizes these results, without disaggregating by type of degree.

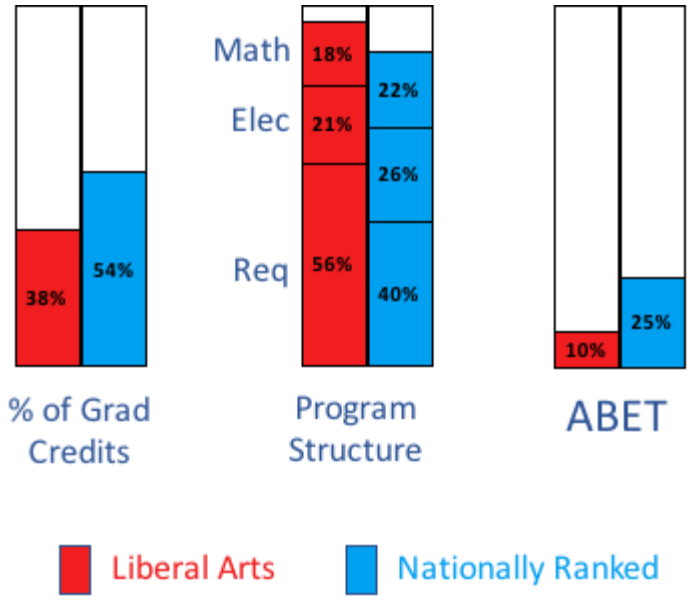

Figure 2. Comparison of computer science program structures between liberal arts institutions and nationally ranked universities.

Computing and Liberal Education

Computing education and liberal education support each other in many ways. Walker and Kelemen [26] identify the synergies, including how liberal education exposes students to multiple areas of and perspectives on knowledge that prepare graduates to understand both developer and client concerns in the computing industry, more generally how even technical employers value liberal arts "soft skills" (e.g., communication, team work, etc.), how computational thinking is becoming an important element of problem solving in all fields, and how anyone who wants to play a role in modern society increasingly needs to understand computing's impact on it. Although Walker and Kelemen frame their discussion in terms of computer science rather than all computing fields, most of their points apply to the other computing fields to at least some degree. Their arguments illustrate both how students become better computing professionals by studying in a liberal education context, and how a modern liberal education is enriched by including computing.

Committee members echoed and amplified these ideas. They confirmed that employers like liberal arts computing graduates, often wanting more once they hire one. Concrete examples of how liberal arts curricula develop desirable soft skills included writing-across-the-curriculum programs, and exposure to other cultures through study abroad. Indeed, our 2016 survey estimated that the median liberal arts computing program sees 20% of its majors study abroad, with at least one respondent reporting a 60% study abroad rate. Others outside the committee also recognize that the humanities and social sciences have a great deal to contribute to conversations about social consequences of computing; for example Keating and Nourbakhsh [16] propose a course that introduces students of artificial intelligence to problems surrounding human power structures and the very meaning of humanness. Computing faculty who have contacts across disciplinary boundaries can play a role in inclusion and equity, for example

working with colleagues in the humanities and social sciences to explore what explicitly anti-sexist and anti-racist curricula in computing could be.

In the other direction, computing has much to offer traditional liberal arts disciplines, and liberal arts curricula have the flexibility to infuse computing into those programs. Liberal education is an ideal context in which to pursue the "CS+X" movement [23] in innovative ways. Computing can also play a significant role in an institution's general education program. For example, Barr et al. [5] and O'Hara et al. [19] discuss approaches to including computer science content in a first-year seminar, while Cliburn [8] describes a CS0 course created to contribute to a college-wide "liberal arts degree requirement."

Incorporating computing into general education makes computing programs more visible than they otherwise would be to students [8]. In this vein, committee members noted that thanks to students' curricular flexibility in liberal arts colleges, liberal arts computing programs attract students with a wide variety of goals and abilities. Some students take multiple courses in computing without intending a major or even minor in the field. While this can enhance intellectual diversity in classes, it requires departments to decide whether to create special introductory and intermediate courses for such students, or to keep everyone on a single course track. It also makes advising and mentoring students challenging, as faculty must avoid making assumptions about students' preparation for computing, career goals, or areas of interest. In all cases, liberal arts computing courses and curricula must be designed to serve students with a wide range of interests and to prepare majors for a broader variety of careers than just the typical technical ones. Liberal arts computing curricula also need to be flexible about their requirements in order to leave room for computing electives and interdisciplinary interests.

While an institution's liberal education mission can create opportunities for computing to transcend traditional disciplinary boundaries, it may also constrain what computing majors can do. Common constraints include limits on the number of courses that can be required [4], or needing to conform to a writing-across-the-curriculum or writing-enhanced-curriculum model. Liberal arts colleges commonly also want students to devote significant time to activities outside their majors.

Committee members also believed that computing programs in liberal arts institutions sometimes struggle to educate non-computing faculty and administrators about what computing is, what it needs, and how it fits into a liberal education. One area where the committee reported particularly important differences with non-computing colleagues was promotion and tenure expectations. Computer science's standards of scholarly achievement differ from those in other fields, particularly in accepting conference presentations as important scholarly contributions in addition to journal articles [20]. On the other hand, computing faculty in interdisciplinary research projects are sometimes expected to provide technical support that does not constitute computing research. While these concerns are not unique to liberal arts colleges, the liberal arts setting can exacerbate them because it is less likely to offer a pool of colleagues with similar scholarly expectations, or to have come to grips with what interdisciplinary computing scholarship looks like.

Other concerns include faculty salaries, needs for research and teaching laboratories, disciplinary affiliation (e.g., whether computing should be considered a science or a professional program), etc. The struggle to balance current expectations of employers and advancing technology with the enduring principles of the field also seems to be more strongly felt in computing than in more traditional liberal arts fields. Nonetheless, Fidoten and Spacco [13] found broad agreement between computer science faculty and faculty in other areas at midwestern liberal arts colleges on the basic activities and kinds of knowledge that characterize computer science, as well as general agreement that computer science is a legitimate subject for liberal arts institutions. While individual institutions and their faculty certainly differ in how well they understand and accept computing, and the details of how it should be implemented within the liberal arts, the general picture appears to be one of broad agreement despite differences in detail.

In accommodating both the opportunities and constraints of liberal education, computing programs in the liberal arts must tailor themselves to their environment and institutional goals rather than trying to replicate what non-liberal-arts programs do. Such adaptation has been a decades-long motivation for a series of model computer science curricula from the Liberal Arts Computer Science Consortium [7]. As seen in these models, a computing program for the liberal arts can be different from non-liberal-arts recommendations while still being a strong program that produces graduates with the skills the world expects from computing education generally. The case studies collected by Baldwin et al. [4], while limited in number, support the idea that liberal arts computer science graduates are competitive or better than competitive with graduates of other programs in the workforce and graduate school.

Small Institutions

While "small college" and "liberal arts college" are not synonymous, many liberal arts computing programs are indeed small programs or located in small colleges. According to the committee's 2016 survey, members' programs graduate a median of 10 students per year, with a range from 1 to 60. Similarly, the median number of faculty advising majors in computing programs is 3, and ranges from 1 to 14. These numbers are broadly similar to those in Baldwin et al.'s case studies [4], where the example programs graduate between 5 and 15 students per year, taught by between 2.5 and 7.5 full-time-equivalent faculty (note that the two studies use different measures of faculty size, namely number advising versus number teaching).

Many of the challenges faced by liberal arts computing programs are thus challenges of small departments in small colleges. Broadly speaking, these challenges center around teaching a wide-ranging and rapidly changing subject with a minimal number of faculty. However, the liberal education philosophy sometimes limits solutions in ways not shared with other small colleges, for example mandating faculty participation in such college-wide programs as a first-year seminar as well as in their own department's programs.

The most obvious consequence of having a small faculty is that it limits the number of course sections a department can offer. This is particularly frustrating for departments that want to, or that are expected to, contribute to general education or interdisciplinary programs as well as to offer a conventional computing major. Some participants in the committee's discussions are unable to offer even required courses every academic year. Students who start a program late may therefore have only one chance to take some of those courses, in turn meaning that oversubscribed

courses cannot simply have enrollment capped in the expectation that students will take them later. As more students want to take a second or third or even later computing course, course capacity becomes a problem beyond just the introductory course(s).

Aggravating course capacity problems, liberal arts computing programs are, with a few exceptions, experiencing the booming enrollments other institutions report. About two thirds (65%) of the respondents to the 2016 survey reported current enrollments above their long-term average, and nearly half (45%) said enrollments were up by a factor of two or more. However, mechanisms for coping with enrollments are different from those at larger institutions. Contributors to committee discussions find it hard to hire part-time faculty to cover extra sections of courses, especially when their institutions are geographically isolated. Undergraduate teaching assistants can help to some extent, but it can be hard to keep the pipeline of qualified assistants full when there are a small number of students to begin with. Moreover, as students with a wide variety of backgrounds and interests become more drawn to computing, departments face increased pressure to offer multiple entry points to their programs, further diluting staffing.

The "simple" solution to staffing shortages, namely hiring and retaining more staff, is of course not at all simple in the computing fields. This is a problem across computing education [18], but is particularly bad for liberal arts colleges. Participants in committee discussions pointed out that small or geographically isolated departments are unattractive to some potential faculty, and, more seriously, that careers in liberal arts colleges are very far from the ones encouraged by the research universities that prepare future faculty. Davis et al. [9] seek to educate graduate students about careers at liberal arts colleges and to confront some of the (mis)perceptions that surround those careers.

Finally, geographic isolation was a topic that appeared several times in committee discussions. While isolation can mean a low cost of living and pleasant natural environment, many discussants also felt professionally as well as geographically isolated. They reported difficulty finding research collaborators, and those who do education research have trouble finding student populations large enough to generate statistically significant results.

While committee discussions tended to focus on drawbacks rather than benefits of small colleges, it is important to realize that small institutions do have some advantages over larger ones. Peterson points out that small colleges tend to be more collegial than larger schools, and are more likely to be communities based on strong interpersonal connections [21]. In a direct application to computer science education, Dickson describes how the more personal atmosphere enriches the experiences of undergraduate teaching assistants by giving them personal responsibility to, and a personal stake in, the professor, course, and students with whom they work [11].

The Focal Questions

The two questions at the heart of the committee's charge are

1. Does the liberal arts computing education community need some sort of organization to support its activities?
2. Does the larger computing education community need some identified organization it can turn to as the voice of liberal arts computing?

Our 2018 survey focused on these questions and they also generated robust, ongoing discussion in committee conference sessions.

Does Liberal Arts Computing Education Need an Organization?

Members of the committee have a passionate desire for a permanent group of liberal arts computing educators. This was clear in the unexpectedly high interest expressed in the committee's work by the liberal arts computing community, and in multiple conference discussions, whose tenor sometimes seemed to be not so much "if we had a group…" as "when we have a group….". As noted earlier, over 100 people follow the committee's mailing list; attendance at an organizational meeting for the committee at SIGCSE 2016 exceeded the capacity of the (large) room reserved for it, and attendance at SIGCSE 2017 and 2018 special sessions and birds-of-a-feathers pushed room capacity. The committee itself occasionally served as a prototype for such a group, for example its mailing list was used at least once to request advice on teaching computing rather than for committee business per se.

Our conference sessions produced a number of ideas about specific ways in which a formal group could support liberal arts computing education, and our 2018 survey also asked about possible group contributions. Ideas from these sources include…

- Facilitating communication between liberal arts computing educators, thereby breaking down some of the isolation of small or geographically isolated departments and allowing them to share information about common concerns such as curricula, course syllabi and content, outreach and advocacy strategies, etc.
- Facilitating research collaborations between liberal arts computing educators at different institutions, and between liberal arts colleges and research universities, including providing or identifying opportunities for liberal arts students to engage in computing research and facilitating publication of their results
- Supporting liberal arts faculty, and even more so students, who want to attend or present at research conferences
- Helping liberal arts computing departments hire and retain faculty by, for example, publicizing creative or particularly effective approaches, developing guidelines for the minimum educational preparation someone hired from a related field should have in order to teach computing, etc.
- Contributing to computing curriculum development efforts to ensure that they reflect curricular ideas from liberal arts colleges as well as from more technical institutions
- Increasing awareness of liberal arts colleges as career choices for graduates of research universities
- Increasing awareness of computer science among colleagues in other fields
- Increasing awareness of the strengths of liberal arts computing programs among potential employers of graduates and among potential students and their families

- Being a collective voice for the needs and interests of liberal arts computing
- Gathering and disseminating data on the state of liberal arts computing education including teaching load, salary, lab and instructional resources, etc.
- Sponsoring conferences, conference sessions, workshops, and similar meetings focused on liberal arts computing education
- Managing other projects in support of liberal arts computing (e.g. outreach, small grants, writing white papers to give to deans, etc).

These contributions to liberal arts computing are better made through a permanent, standing group than through a series of independent, one-time projects. Some of the contributions, such as advocacy and assistance programs, require continuous ongoing efforts. Others, such as collecting data and sponsoring conferences, should be periodically recurring activities. The sheer number of forms of support the liberal arts computing community desires makes coordination between them imperative. On all of these counts, a permanent group is preferable to one-off projects.

Does Liberal Arts Computing Education Need a Voice?

There is considerable sentiment within the liberal arts computing education community that it needs a voice to communicate its interests to outside groups. Many of the activities suggested for a liberal arts computing group reflect this sentiment. That sentiment was also clear in the 2018 survey, where 61 of 63 respondents reported engaging in some sort of advocacy for liberal arts computing education to outside constituents and 58 respondents reported engaging in advocacy with more than one population. Comments on both surveys and in email discussions also express this view, for example

> "We have unique qualities that sometimes need to be communicated to parties outside the community" (email post, July 2017).

> "I think CS is crucial to liberal arts education, but it is difficult sometimes to help non-CS and non-science people understand what CS is." (survey response, spring 2016)

While respondents to our 2018 survey had almost all advocated for liberal arts computing with others, there was a distinct difference between the groups they were able to talk to and the ones they felt needed to hear the message. Figure 3 summarizes the differences by showing the top four populations (plus an "other" population) that respondents reported engaging with and the top four populations with which they desired to engage. The only group in both sets is prospective students and their parents. Otherwise, actual advocacy tends to be internal, in that it engages students, faculty, and staff in respondents' own institutions, or at least other academics outside respondents' institutions. In contrast, groups that respondents wanted to reach were more external, e.g., employers, graduate schools, or policy-making bodies. A body that was the organized representative of liberal arts computing education could presumably reach these populations more effectively than individuals acting on their own can.

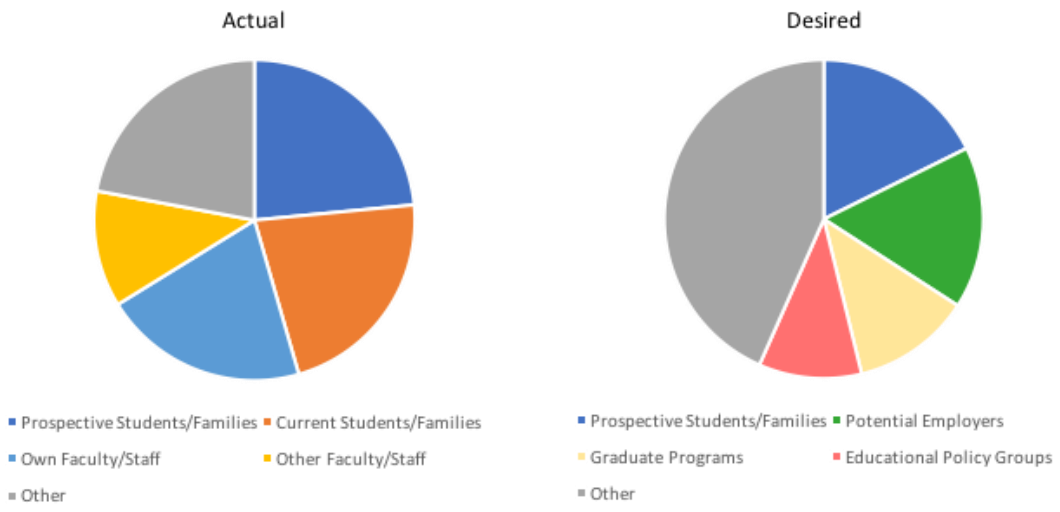

Figure 3. Top groups that liberal arts computing educators actually advocate with compared to groups they want to advocate with.

One third of the respondents to the 2018 survey (35%) reported having personally represented liberal arts computing education in some larger context. Commonly mentioned contexts included conferences (both inside and outside of computing education), publications, and curriculum committees or task forces, among others. These responses suggest that non-liberal-arts groups are also interested in liberal arts computing's perspective. Thus, in addition to liberal arts computing educators wanting a voice to represent their interests to others, it seems the larger community would also benefit from an identifiable interlocutor on issues affecting liberal arts computing.

Recommendation and Conclusion

The committee found a large community passionate about computing as a part of liberal education, and about liberal education as an environment in which to teach computing. Through conversations with this community, surveys, and a review of the literature, the committee has identified a number of ways in which computing educators can tap liberal education principles for the mutual benefit of both computing and liberal education. We also found some tensions arising from perceptions of computing by others in the liberal arts, or from perceptions of the liberal arts by others in the computing community, but these are matters of perception more than irreconcilable differences between computing and liberal education. Liberal arts computing programs often differ from programs at top-ranked national universities in, for example, size, requirement structure, degree granted, and interest in ABET accreditation, but nonetheless deliver strong and effective computing education.

Based on these findings, the committee makes one main recommendation and one subsidiary one:

1. (Main) That an organization be created to support and represent liberal arts computing educators.
2. (Subsidiary) That a small group immediately begin identifying and implementing specific ways to make the liberal arts computing organization a reality.

Liberal arts computing education is a sufficiently distinct subfield of general computing education to justify some sort of organization for its practitioners; the committee's members are adamant about their desire for such an organization. We absolutely do not, however, suggest that the new organization be separate from existing computing education groups—liberal arts computing education is, after all, still computing education. Ideally, the new organization could somehow be an entity within, or at least closely affiliated with, such groups as SIGCSE or the Consortium for Computing Sciences in Colleges. We also strongly believe that the new organization should follow the committee's lead in taking a broad view of "liberal arts," concentrating on how the philosophy of liberal education impacts computing education and vice versa more than on how certain disciplines or kinds of institution do it; the organization should be open to all who are interested in these issues. Within these parameters, the new organization will serve two broad purposes:

- Providing a framework within which the liberal arts computing community can support and strengthen itself, e.g., organizing meetings and virtual spaces in which members of the community can network, conducting and publishing studies of the community, etc.
- Representing the liberal arts computing community to others, e.g., working to ensure accurate perceptions of liberal arts computing among employers, graduate schools, potential faculty, and future students; providing a liberal education voice on curriculum task forces and similar bodies; etc.

The recommended organization will not, of course, simply spring into existence because the committee says it should. The second recommendation is therefore that someone start acting now, while interest is still high, to create that organization. "Someone" could well be a small subset of the existing committee, and the committee leaders are beginning to identify volunteers. In any case, actions that need to be taken include identifying and approaching potential partner organizations, prototyping specific activities that the new organization might sponsor (perhaps initially as workshops, birds-of-a-feather, or similar events within existing meetings), and developing a budget and revenue stream for the new organization.

Acknowledgements

We are deeply grateful to the 100+ people who joined the committee through its mailing list, and who contributed to our discussions, both face-to-face and electronic. Special thanks to Henry Walker for his tireless review of drafts of this report. We also thank the SIGCSE Board for creating the committee and supporting its work, and the 2017 and 2018 SIGCSE Symposia for hosting our sessions. The findings in this report are the authors' and the committee's, and do not reflect any official positions of SIGCSE or its Symposia.

References

1. Association of American Colleges and Universities. "What Is a Liberal Education?" https://www.aacu.org/leap/what-is-a-liberal-education. Accessed 2018 June 4.
2. Baldwin, D. Draft report of the SIGCSE committee on computing education in the liberal arts. in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. (Baltimore, Maryland, USA: ACM, 2018), 1047 - 1048.

3. Baldwin, D., Braught, G., and Holland-Minkley, A. Computing education in liberal arts colleges: a status report of the SIGCSE committee. in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, (Seattle, Washington, USA: ACM, 2017), 675 - 676.
4. Baldwin, D. *et al*. Case studies of liberal arts computer science programs. *ACM Transactions on Computing Education*, 10, 1, (March 2010), doi:10.1145/1731041.1731045.
5. Barr, V., *et al*. Computer science topics in first- and second- year seminar courses. in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. (Seattle, Washington, USA: ACM, 2017), 643-644.
6. Beidler, J., Austing, R.H., and Cassel, L.N. Computing programs in small colleges. *Communications of the ACM*, 28, 6, (June 1985), 605 - 611.
7. Bruce, K., Cupper, R., and Drysdale, R.L.S. A history of the liberal arts computer science consortium and its model curricula. *ACM Transactions on Computing Education*, 10, 1, (March 2010), doi:10.1145/1731041.1731044.
8. Cliburn, D.C. A CS0 course for the liberal arts. in *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education*, (Houston, Texas, USA: ACM, 2006), 77 - 81.
9. Davis, J. *et al*. Computer science faculty careers at liberal arts colleges. *XRDS: Crossroads, The ACM Magazine for Students*, 21, 3, (Spring 2015), 13 - 15.
10. Dewey, J. *et al.* The function of the liberal arts college in a democratic society. *The American Scholar*, 13, (October 1944), 391 - 406.
11. Dickson, P.E. Using undergraduate teaching assistants in a small college environment. in *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education*. (Dallas, Texas, USA: ACM, 2011), 75 - 80.
12. Dinerstein, N. Does computer science belong in a liberal arts college? *ACM SIGCSE Bulletin*, 7, 2, (June 1975), 55 - 64.
13. Fidoten, H. and Spacco, J. What do computer scientists do? a survey of CS and non-CS liberal arts faculty. in *Proceedings of the 2012 Conference on Innovation and Technology in Computer Science* (Haifa, Israel: ACM, 2012), 279 - 284.
14. Gibbs, N. and Tucker, A. A model curriculum for a liberal arts degree in computer science. *Communications of the ACM*, 29, 3, (March 1986), 202 - 210.
15. Joint Task Force on Computing Curricula. *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*. (New York, New York, USA: ACM and IEEE Computer Society, 2013)
16. Keating, J. and Nourbakhsh, I. Teaching artificial intelligence and humanity. *Communications of the ACM*, 61, 2, (February 2018), 29 - 32.
17. Liberal Arts Computer Science Consortium. A 2007 model curriculum for a liberal arts degree in computer science. *ACM Journal of Educational Resources in Computing*, 7, 2, (June 2007), doi:10.1145/1240200.1240202.

18. National Academies of Sciences, Engineering, and Medicine. *Assessing and Responding to the Growth of Computer Science Undergraduate Enrollments*. (Washington, DC, USA: The National Academies Press, 2018).
19. O'Hara, K.J., Burke, K., Ruggiero, D., and Anderson, S. Linking language & thinking with code: computing within a writing-intensive introduction to the liberal arts. in *Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education* (Bologna, Italy: ACM, 2017), 269-274.
20. Patterson, D., Snyder, L., and Ullman, J. Evaluating computer scientists and engineers for promotion and tenure. https://cra.org/resources/best-practice-memos/evaluating-computer-scientists-and-engineers-for-promotion-and-tenure/. Accessed 2018 June 28.
21. Peterson, H.L. *Leading a Small College or University: A Conversation that Never Ends*. (Madison, Wisconsin, USA: Atwood Publishing, 2008).
22. Roth, M. *Beyond the University: Why Liberal Education Matters*. (New Haven, Connecticut, USA: Yale University Press, 2014).
23. Ruff, C. Computer science, meet humanities: in new majors, opposites attract. *The Chronicle of Higher Education*, January 28, 2016.
24. Streib, J. and White, C. A survey of computer science curricula at liberal arts colleges: a pilot study. *Journal of Computing Sciences in Colleges*, 18, 1, (October 2002), 36 - 42.
25. US News and World Report. National university rankings. https://www.usnews.com/best-colleges/rankings/national-universities?major=Computer+Science. Accessed 2017 February 21.
26. Walker, H.M. and Kelemen, C. Computer science and the liberal arts: a philosophical examination. *ACM Transactions on Computing Education*, 10, 1, (March 2010), doi:10.1145/1731041.1731043.