Enabling Interdisciplinary Instruction in Computer Science and Humanities

An Innovative Teaching and Learning Model Customized for Small Liberal Arts Colleges

William B. Crum, Jr.[0000−0002−0860−4541], Aaron Angello[0000−0003−0245−2169], Xinlian Liu[0000−0002−6945−0750], and Corey Campion[0000−0002−0788−7324]

Hood College, Frederick MD 21701, USA
{crum,angello,liu,campion}@hood.edu

Abstract. Infiltration of data-driven computational methods of humanities research has generated mutual interests between the two communities of computer science and humanities. Larger institutions have adopted drastic structural reforms to meet the challenges to bridge the two fields. Successful examples include the integrated major programs launched at Stanford University and the collaborative workshop at Carnegie Mellon University. These types of exploratory experiments require 1) intensive resources as well as 2) strong support of faculty and administration. At a small college, both can be luxuries. We present an innovative model to carry out effective synchronized courses of computational humanities and digital humanities that pulls together efforts between two small programs and needs little additional support. This paper reviews the proposal, design, and delivery of a pair of interdisciplinary graduate courses in the small college setting. We discuss the details of our implementation and provided our observations and recommendations.

Keywords: Interdisciplinary · Synchronized Courses · Computational Digital Humanities · Interdisciplinary Project-Based Learning · Teaching Pedagogy

1 Humanities in the Age of Big Data

Over the past few years, drastically increased data-collecting capability and massively parallel data processing capability have enabled a swift shift in the way research projects are conducted in humanities, arts, and social sciences fields. The emergence of large data collections, e.g., digitized documents including books, government records, images, audio, and video, websites, social media, health, business, and communication records, sensor data, etc., changes the way scholars generate and present meaningful research in the 21st century [7]. Due to the nature of these large corpora, traditional means of analysis and presentation are no longer sufficient. This research requires research teams with expertise in multiple fields. The trend has been recognized by academia, as higher education

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

institutions are taking steps to offer students interdisciplinary programs which integrate computational and humanities curricula.

Despite the success of such programs in leading institutions such as The *CS+X* Joint majors at the Stanford University and a *Conjoint Model* piloted at Villanova University [21], it remains a challenge in smaller colleges to offer similar programs because of barriers in institutional support, faculty buy-in, and curricula design [22].

Among the challenges confronting liberal arts education today is a fundamental disconnect between the curricula that many institutions offer and the training that many students need. In the early twenty-first century, most colleges and universities still adhere to the model of disciplinary-specialization that developed in the nineteenth century when the pressures of industrialization and globalization led to the expansion of higher education and a need to justify the growing number of tenure lines within an institution. Though an important asset for students and faculty alike, discipline-specific models of higher education struggle to prepare students for the kinds of interdisciplinary collaborations now expected by many employers. Aware of the need for change, many institutions have tried to translate the widespread rhetoric about interdisciplinarity into new programs and curricula that better serve today's students.

With an eye toward providing students in the Master of Arts in Humanities and Master of Science in Computer Science programs the opportunity to practice the interdisciplinary collaboration, the program directors and co-instructors undertook to create an exciting new learning opportunity within the unique budgetary and enrollment environment found at the hundreds of smaller institutions across the United States. From the identification of student learning outcomes and the development of an interdisciplinary syllabus to the logistics of sharing a course across programs and the experiences of both instructors and students, the paper considers both the benefits and challenges of such collaboration and offers recommendations for those looking to undertake similar projects at their own institutions.

2 Design and Architecture of the Courses

2.1 Customization Based on Local Resources

The project began when the director of the M.S. in Computer Science program approached the director of the M.A. in Humanities program about the possibility of offering humanities courses that would enhance the communication skills of computer scientists and help them develop the empathy required to understand the human communities their computer programs serve. Coincidentally, the M.A. director had been looking for ways to help humanities students expand their research and share their ideas with a wider audience through the use of technology. The results of this exchange were the institution's first-ever paired-courses in digital humanities. Unlike traditional dual-listed courses, the paired courses allowed students to explore digital humanities within their own

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

disciplines before coming together to collaborate on shared projects. Students from both fields learned to balance pragmatism and ambition through collaboration and communication. By combining enhanced training in the students' fields with the opportunity for interdisciplinary collaboration, the courses produced an experience that more closely approximated the working environments of today.

To mitigate the barriers to implementing a joint program at a small college, we came to the realization that our approach should:

1. take little additional resources
2. require few curricular changes to existing programs
3. provide maximum support to the instructors

To gain institutional support, we used recent research findings [20], [15], [12] and media coverage [16],[3] to reason with the administrators that Computer Science could be the *enabler* of liberal arts education, which is the core of our College's education missions.

Typically, computer science students are given assignments related to the specified course in isolation. Even when teamwork is assigned, the teams are made up of computer science students. Neither the project goals nor the team makeup allows students to gain experience in interdisciplinary research or collaboration beyond the field of computer science. The same is often true for graduate students in the humanities, few of whom enjoy opportunities to collaborate with and benefit from the unique skill sets of their colleagues in the natural and social sciences.

As we went about planning the two courses, we confronted sets of problems unique to each group of students. Very simply, most of the students who were a part of the Masters in Humanities program had few technical skill, while the students who were a part of the Masters in Computer Science lacked an understanding of the kinds of research undertaken in humanities fields. These concerns had to be addressed separately, but we also had to keep in mind that one of our primary goals was to bring these two classes together.

The students in the Masters of Humanities program had virtually no previous experience with digital humanities (DH). For the most part, these students are high school teachers or adults who are lifelong learners, interested in expanding their knowledge of topics in the humanities in general. They are not specialists, and have not spent time in research universities with digital humanities programs. There was a general sense of curiosity among the students, a curiosity that presumably inspired them to sign up for the class, but they demonstrated a distinct lack of understanding of the history of DH, of the types of projects that are or ought to be considered DH, or even of what DH, broadly speaking, is.

Furthermore, writing code was a terrifying prospect to most of these students. A few had had some web development experience, but none were comfortable even with HTML or CSS. We had to assure the students that practicing DH does not necessarily mean one has to write code (although there are certainly those who have argued otherwise [14]), that our very intention in setting the classes up the way we had was to create an environment in which computer programmers

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

and humanists could collaborate. Different people have different skill sets, and digital humanities are, in no small part, about collaboration. The students would not need to learn to write code in this class, but they would need to have some understanding of how the code works so that they could communicate with the computer programmers.

2.2 Student Learning Outcomes

Upon completion of this course pair, students were able to:

- Demonstrate critical technical skills necessary to conduct interdisciplinary research in computational social sciences and digital humanities.
- Demonstrate the skills necessary to work in interdisciplinary teams.
- Demonstrate effective programming skills necessary to develop natural language processing applications in the humanities (for students attending the Computational Humanities class).
- Speak intelligently regarding topics in computational humanities (for students attending the Computational Humanities class).
- Recognize limitation of technologies and technical methods (for students attending the Digital Humanities class).

2.3 Course Planning

We began the course by trying to define the term “digital humanities.” As anyone who works in the field knows, this is a daunting prospect. Even among the field’s preeminent practitioners, there is much debate about what this term means. The students read a number of “defining DH” articles, including Matthew Kirschenbaum’s “What is the Digital Humanities and What’s It Doing in English Departments?” [8] and Patrik Svensson’s “The Landscape of Digital Humanities,” [17] both articles that have practically become canonical in Introduction to DH classes. We also found that Svensson’s “Beyond the Big Tent” [18] helped the students engage in the “what counts as DH” debate. For our purposes, when it comes to the “who’s in and who’s out” debate [14], we found it useful to try to embrace the broadest and most inclusive definition of DH. We tried to adopt Svensson’s recommendation that we think of DH not as a “big tent,” a field in itself that defines itself by being exclusionary, but rather “as a meeting place, innovation hub, and trading zone,” a site that highlights a “commitment to interdisciplinary work and deep collaboration” [18]. Given the students’ backgrounds in English, history, and art, and given their trepidation with regard to digital research methods in the humanities, we found that thinking about DH in this way allowed for more fruitful engagement in the class.

Next, we provided a bit of historical contextualization for the field. Students read accounts of early examples of the use of computing in the humanities such as Father Roberto Busa’s twenty-five-year project creating a lemmatized concordance of all eleven million words in the works of St. Thomas Aquinas’ writings, a project which began in 1949 [1]. The students went on to look at the work of

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

Vannevar Bush's "As We May Think" [2] and Theodor Nelson's "A File Structure for the Complex, the Changing, and the Indeterminate" [6] so that they could understand the origins of linked file structures which are so ubiquitous today.

The bulk of the class, however, was focused upon making. After analyzing a number of existing projects, the students were able to get hands-on experience in the areas of text analysis (e.g. Voyant Tools), creating digital collections (Omeka), encoding metadata (XML and TEI), network visualization (Gephi and Palladio), and geospatial humanities (CartoDB). The idea was to give the students an overview of some common, powerful tools used in a number of successful research projects that are also relatively unintimidating to the DH novice. Once they had an idea of what was possible, they began to conceive projects related to their particular areas of interest, and they began to work on those projects.

In Computational Humanities (CH), the students were taught natural language processing (NLP) in Python using the Natural Language Toolkit (NLTK) [9]. The only prerequisite was a programming class in Java or Python. Only one student had taken a course in Python. The students in CH were traditional age international students largely from one country. These students had no background in natural language processing or in digital humanities.

This project-based course involved three strands: the Python language, NLTK functions, and natural language processing methods. During the early stage, the instruction interlaced these strands with emphasize on Python language and basic NLTK functions. Initially, instruction in natural language processing focuses on basic concepts and vocabulary. Once students gained a firm footing in the Python language, the students' attention turned primarily to the use of NLTK to investigate various aspects of natural language processing.

Where these classes differed from most other introductory courses, however, was in their emphasis on collaboration across disciplines. These courses ran concurrently. Early in the semester, and again in the middle of the semester, the professors from each class met with the students from the other class. The humanities professor prepared lectures and discussions aimed at helping the computer science students better grasp some of the conceptual and methodological approaches to research in the humanities and its importance. The computer science professor introduced the humanities students to the basics of programming and natural language processing using Python and the Natural Language Toolkit (NLTK). Many of the computer science students had spent their entire academic careers as far from English, art, history, and philosophy as they could, opting to surround themselves with other programmers and "math-brained" people. Similarly, the humanities students, in many cases, winced at the prospect of learning to code. They had simply never been exposed to it.

Both classes did find value in their exposure to these other disciplines, but the real value of running these classes together, and the explicitly stated goal of organizing it this way, was to provide an opportunity for the students to learn to communicate to each other across disciplines, to learn to speak the language of their counterparts.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

3 Implementation

Two graduate courses Computational Humanities (Computer Science Department) and Introduction to Digital Humanities (Humanities Department) were run in parallel.

Each course was a normal one semester, 15-week course meeting once a week. Originally, we planned to run the courses on the same day and time. However, due to scheduling issues, the classes ran on the same day but during overlapping time periods. The overlap was during the last half of the Digital Humanities course and the first half of the Computational Humanities course.

During the first three weeks, each course met independently. Students in the Digital Humanities course read articles about the history of humanities computing and about ongoing discussions and debates in the field. They discussed foundational texts, including Vannevar Bush's "As We May Think" [2], Theodor Nelson's "A File Structure for the Complex, the Changing, and the Indeterminate," [6], and the work of Father Roberto Busa [1]. Additionally, they read and discussed a number of prominent articles that address the question of how we are to define the digital humanities [18] [8], [14]. During this initial three weeks, the students were able to begin situating themselves within the complex field of digital humanities and to start grasping a fundamental understanding of the kinds of work that go in the field. Computational Humanities students began to learn the fundamentals of Python programming and the use of NLTK functions to perform basic natural language processing activities. They developed programs that performed basic activities, such as accessing text corpora, text from the Internet, and other lexical resources, extracting content from web pages, processing raw text, searching text, counting vocabulary, using regular expressions to detect word patterns, creating frequency distributions, and plotting and tabulating the results of frequency distributions. By the end of the first three weeks, students developed skills to write programs in Python and to use NLTK functions. They gained a basic understanding of the vocabulary and concepts of natural language processing.

On the fourth week, the classes met together for part of the session. During this time, the students got to know each other and discuss some preliminary project ideas. Following the joint meeting, the Computational Humanities students met with the humanities professor to begin understanding the myriad ways humanists are using digital tools to expand research opportunities and to begin understanding the value of employing computational methods in humanities research. The Introduction to Digital Humanities students met with the computer science professor to receive some basic instruction in writing Python programs in natural language processing.

During the next three weeks, each course met independently. The Digital Humanities course began to narrow its focus, looking at specific areas of focus, methods, and tools. They began looking at the markup, metadata, and the value of TEI (Text Encoding Initiative). They also read Franco Moretti's *Maps, Graphs, and Trees* [11] and began exploring the potentials of data visualization, both theoretically [10], [13], [4] and practically, by creating their own small-scale

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

projects using Gephi and Voyant tools. With experience in writing programs that perform basic natural language processing activities, the Computational Humanities students were ready for more advanced activities. Students wrote programs using part-of-speech taggers to process sentences into lists of (word, part-of-speech) pairs and using tagged corpora and Python dictionaries to analyze text. Students wrote programs to evaluate the performance of automatic taggers. Through this work, students gained an appreciation for the processes and problems associated with categorizing and classifying text.

On the eighth week, the classes met together for part of the session. During this time, the students formed research teams and began work on the projects. Following the joint meeting, the Computational Humanities met with the humanities professor to receive additional instruction in digital humanities, this time exploring a number of existing DH projects, while the Introduction to Digital Humanities met with the computer science professor to receive additional instruction in writing Python programs in natural language processing.

During the next four weeks, each course met independently and the interdisciplinary teams formed during week eight worked on the final project outside of class. The Digital Humanities students examined geo-spatial humanities projects and read articles dealing with potentially new ways of presenting scholarly research [5], explored issues related to digital pedagogy, and discussed issues associated with racial and gender identities in the field. In the past, Computational Humanities students examined text at the word level. Now, the students' focus changed to extracting meaningful word groups, such as noun phrases from the text. They wrote programs that analyzed sentence structure and built and parsed simple grammars.

During the last three weeks, the classes met jointly to work on team projects and give presentations of the projects.

4 Outcome and Discussions

To assess the effectiveness of our approach, we assessed students' readiness in terms of grasping of required skill sets to conduct interdisciplinary projects in computational digital humanities. We then estimated the metrics again towards the end of the class. We provided these figures along with the average grades of the joint project and students' semester grade in Table 1.

The semester grade includes non-interdisciplinary projects given throughout the semester as well as the interdisciplinary joint project. Joint project grades are team-based. The instructors met to jointly evaluate each team project. The joint project evaluation criteria include:

1. quality as a digital humanities project
2. the extent was it interdisciplinary
3. quality of the team work
4. progress towards completion as a proof of concept
5. quality of the grant proposal
6. class presentation

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

		grasping of required skill sets		average grades received	
		pre-class	post-class	joint project	semester grade
CS students	NLP skills	20%	80%	96%	91%
	HUM skills	10%	60%		
HUM students	NLP skills	15%	75%	96%	95%
	HUM skills	50%	85%		

Table 1. Student's readiness on skill sets required to succeed in an interdisciplinary project in computational humanities before and after the joint class, and the average grades they received in the joint project as well as the average semester grade.

4.1 Observations

This was a first step in the development of an interdisciplinary program involving the Computer Science and Humanities departments. On the whole, it was a good first step.

Due to limited time, it was understood that students would be unable to deliver a production-level project. Therefore, students were to make sufficient progress to provide proof of concept and write a grant proposal.

The projects created by the teams were generally satisfactory. There were some issues regarding team interactions. However, they were manageable. Some projects tended to lean more towards the digital humanities side, while others tended to lean more towards the natural language processing side.

One project that seemed to really get the right mix was a team comprised of an art teacher and a computer science student. The team wanted to create an interactive web-based art history timeline. To start, they wanted to transfer the art teacher's 250 item art history PowerPoint presentation to a web page. The art teacher's approach was to copy and paste materials from the PowerPoint presentation. However, the computer science student saw that this process could be automated via a natural language processing program.

He developed programming tools to extract information from the PowerPoint presentation and insert the information into an Excel spreadsheet. The goal was to create a database to use in the construction of a web page.

In developing the program, they discovered the difficulty of writing a program dealing with the complexities for natural languages. For example, the program needed to extract date information. Date information can take many formats, e.g., numerical (100 BCE) or verbal (sixth century), which makes it more difficult to extract. They overcame many obstacles. However, due to limited time, they were unable to solve all the problems by the end of the semester.

4.2 Issues and Recommendations

Scheduling While the class schedules overlapped, they did not meet at the same time. This meant that during weeks four and eight the time available for both the join meeting component and cross-training was limited to about two-thirds of the normal class time. In reality, both groups needed more time for cross

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

training. The Computation Humanities students only had time to get a basic understanding of the vocabulary and concepts of digital humanities. Likewise, the Digital Humanities students only got a chance to see the natural language processing programs. They did do a little "follow the leader" programming with the professor.

Recommendation: Future course offerings of an interdisciplinary nature should meet at the same days and times.

Cross Training The scheduling issue limited cross training. Cross training provides a framework for team members to understand their colleagues in other disciplines. In this setting, the Computational Humanities students needed a basic understanding of digital humanities and the types of research it involves. Likewise, the Introduction to Digital Humanities students needed a basic understanding of writing natural language processing programs. With crossing training, team members can more effectively communicate and develop a better solution.

Recommendation: Cross training session should increase both in length and number. Concurrent meeting times should help.

Team Meetings Many students in Introduction to Digital Humanities are adult learners with full-time jobs. This makes team meetings outside of class difficult.

During the last three weeks of the courses, the Computational Humanities students made an effort to meet together during the Introduction to Digital Humanities scheduled time. However, some were not able to make the beginning of the class.

Recommendation: Concurrent meeting times should help provide more opportunities for team meetings including additional out-of-class meeting time before and after regular class meetings.

Student Mindsets Just because students know a class will have an interdisciplinary project does not mean they will want to work in groups. Students sign up for courses for all kinds of reasons.

Students felt most comfortable working with colleagues within their own discipline. Some students did not want to work in teams at all. Therefore, it was necessary to require interdisciplinary teams and to provide guidance to those reluctant students to work with students of a different discipline.

Many of the digital humanities students, i.e., teachers, were experienced in organizing and presenting information and in leadership. In some cases, they did not have the patience or the inclination to delegate tasks to "junior" members of the teams. Likewise, "junior" members did not always demand a greater role in the project. Therefore, it was necessary to provide guidance for students dealing with team dynamics.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

Recommendation: Concurrent meeting times should help provide more opportunities for students to get to know their counterparts in the other class. The opportunity for great human connection should help change negative mindsets.

Projects The final project was the only interdisciplinary project.

Recommendation: Ideally, the development of project ideas and formation of teams should occur earlier in the semester. This allows more time to complete the project and develop the skills necessary to work in interdisciplinary teams.

The development of one or two mini-interdisciplinary projects early in the course could be beneficial to develop skills necessary to work in interdisciplinary teams. This could be professor directed, i.e., the professors choose project topics, scope, and teams.

5 Acknowledgement

We thank the Hood College Graduate School and our corresponding departments for their support. We are indebted to our students who embraced this pilot project and worked with us along the way when we rethought, re-planned, and refined the project.

This project is in part supported by a grant from the National Endowment for the Humanities. The computing resources are made available in part by Extreme Science and Engineering Discovery Environment (XSEDE) [19], which is supported by National Science Foundation grant number ACI-1548562 under Educational Allocation ASC140025.

6 Conclusion

By combining enhanced training in the students' fields of study with the opportunity for interdisciplinary collaboration, the courses produced an experience that more closely approximated the working environments in which many are now employed. The pilot was well accepted by our students from both Computer Science and Humanities sides. We also learned valuable lessons that will help us to improve this model as an effective teaching strategy.

References

1. Busa, R.: The annals of humanities computing: The index thomisticus. Computers and the Humanities **14**(2), 83–90 (1980)
2. Bush, V.: As we may think. The Atlantic Monthly **176**(1) (1945)
3. Davis, J., Albrecht, J., Alvarado, C., Chen, T.Y., Lee, S.: Teaching and Research in Computer Science at Liberal Arts Colleges: Myths and Reality. Tech. rep., http://www.forbes.com/2006/07/28/leadership-

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

4. Drucker, J.: Humanities approaches to graphical display. DHQ: Digital Humanities Quarterly **5**(1) (2011)
5. Fitzpatrick, K.: Planned Obsolescence: Publishing, Technology, and the Future of the Academy. New York University Press (2011)
6. H. Nelson, T.: Complex information processing: A file structure for the complex, the changing and the indeterminate (01 1965). https://doi.org/10.1145/800197.806036
7. Khan, N., Yaqoob, I., Hashem, I.A.T., Inayat, Z., Ali, W.K.M., Alam, M., Shiraz, M., Gani, A.: Big Data: Survey, Technologies, Opportunities, and Challenges, url = http://www.ncbi.nlm.nih.gov/pubmed/25136682 http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=pmc4127205. The Scientific World Journal **2014** (2014). https://doi.org/10.1155/2014/712826
8. Kirschenbaum, M.G.: What is digital humanities and whats it doing in english departments? ADE Bulletin (150), 55–61 (2010)
9. Loper, E., Bird, S.: Nltk: The natural language toolkit. In: Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1. pp. 63–70. ETMTNLP '02, Association for Computational Linguistics, Stroudsburg, PA, USA (2002). https://doi.org/10.3115/1118108.1118117, https://doi.org/10.3115/1118108.1118117
10. Manovich, L.: What is Visualization. paj:The Journal of the Initiative for Digital Humanities, Media, and Culture **2**(1) (dec 2010), https://journals.tdl.org/paj/index.php/paj/article/view/19
11. Moretti, F., Piazza, A.: Graphs, Maps, Trees: Abstract Models for a Literary History. Verso (2005)
12. Pulimood, S.M., Pearson, K., Bates, D.C.: A study on the impact of multidisciplinary collaboration on computational thinking. In: Proceedings of the 47th ACM Technical Symposium on Computing Science Education. pp. 30–35. SIGCSE '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2839509.2844636, http://doi.acm.org/10.1145/2839509.2844636
13. Ramsay, S.: In praise of pattern. Faculty Publications, Department of English (57) (2005), http://digitalcommons.unl.edu/englishfacpubs/57/, last accessed 15 February 2019
14. Ramsay, S.: Who's in and who's out. In: Melissa Terras, J.N., Vanhoutte, E. (eds.) Defining Digital Humanities: A Reader, pp. 239–241. Taylor and Francis Group, Burlington, VT (2013)
15. Stozhko, N., Bortnik, B., Mironova, L., Tchernysheva, A., Podshivalova, E.: Interdisciplinary project-based learning: Technology for improving student cognition. Research in Learning Technology (2015). https://doi.org/10.3402/rlt.v23.27577
16. Straumsheim, C.: Computer Science as Liberal Arts 'Enabler' (feb 2016), https://www.insidehighered.com/news/2016/02/23/liberal-arts-colleges-explore-interdisciplinary-pathways-computer-science
17. Svensson, P.: The landscape of digital humanities. Digital Humanities **4**(1) (2010)
18. Svensson, P.: Beyond the big tent. In: Gold, M.K., Klein, L.F. (eds.) Debates in the Digital Humanities. University of Minnesota Press, Minneapolis, MN (2012)
19. Towns, J., Cockerill, T., Dahan, M., Foster, I., Gaither, K., Grimshaw, A., Hazlewood, V., Lathrop, S., Lifka, D., Peterson, G.D., Roskies, R., Scott, J.R., Wilkins-Diehr, N.: Xsede: Accelerating scientific discovery. Computing in Science & Engineering **16**(5), 62–74 (Sept-Oct 2014). https://doi.org/10.1109/MCSE.2014.80, doi.ieeecomputersociety.org/10.1109/MCSE.2014.80

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31

20. Walker, H.M., Kelemen, C.: Computer science and the liberal arts: A philosophical examination. Trans. Comput. Educ. **10**(1), 2:1–2:10 (Mar 2010). https://doi.org/10.1145/1731041.1731043, http://doi.acm.org/10.1145/1731041.1731043
21. Way, T., Whidden, S.: A parallel, conjoined approach to interdisciplinary computer science education. In: Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education. pp. 363–363. ITiCSE '16, ACM, New York, NY, USA (2016). https://doi.org/10.1145/2899415.2925486, http://doi.acm.org/10.1145/2899415.2925486
22. Zhang, C.: Interdisciplinary Teaching and Research: Challenges and Solutions. In: Proceedings of the 2017 7th International Conference on Education, Management, Computer and Society (EMCS 2017). pp. 160–163. Atlantis Press, Paris, France (mar 2017). https://doi.org/10.2991/emcs-17.2017.31, http://www.atlantis-press.com/php/paper-details.php?id=25876027

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22750-0_31