Lab Culture versus Hackathons

Departmental computing labs, once the enabler of peer learning, socialization and informal learning among students, seem to have given way to hackathons. Is that a good thing?

Lab Culture

Creating an environment where students informally gather to work on their coursework used to be an important component in computing departments. With students owning their own computers and laptops much of the practice of learning computing for students has moved out of the department labs. Students find other venues like dorm rooms, lounges, cafes, etc. to do most of the programming for their courses. Consequently, the design of computing labs has changed in several college and university departments. These days, computing labs provide a minimal number of workstations since students tend to bring their own laptops on which to do much of the coursework. Work carried out in department labs has been reduced to supervised course-related closed labs that are just another mode of formal instruction. Outside of scheduled course-related labs, students only venture into a lab when they need to see teaching assistants for consulting with issues on their homework.

How students learn computing has also been evolving. Not only have we redesigned the physical lab space to accommodate more students with laptops, we are also using programming environments and tools that support an increased amount of remote work via web-based cloud computing, online interactive eBooks browser embedded repls (read-eval-print-

loop), Jupyter-style online notebooks, etc. While we're quick to adapt and adopt new tools and technologies in our courses, do we also pause to think about their impact on overall learning? For example, students' cognitive models for editing, compiling versus interpreting will be very different when programming using web-based or Jupyter-like sandboxes.

Consequently, beyond scheduled supervised sessions, the labs do not get much use. Lack of presence in the labs outside of scheduled times leaves little room for informal peer learning and social interactions among students. This is a very different picture from how most of us learned computing.

The decline of "lab culture" among students should be a cause for concern. The lack of a supportive informal lab culture raises the question of whether this is something a department can or needs to address. Of course, left unchecked, lab culture can also lend itself to a socially toxic environment—especially when enrollments

are out of balance for gender, race, and ethnicity—that deters students from going to the labs outside of schedule times.

Departments need to make deliberate efforts to create supportive and inclusive environments. Labs used to be a place for students to congregate, interact, and learn from each other outside of scheduled classes. A recent study also asserted that study groups, tutoring sessions, and TA-led help sessions in labs are helpful in encouraging informal learning [9].

Hackathons: Corporate & Collegiate

"...they offer a baptism by fire: a short, intense plunge that assaults the senses and allows employees to experience creative disruption in a visceral way." [1]

Hackathons started in the computing industry as an outlet for passion projects for employees. Software developers would get together for one or two days

over a weekend, pitch ideas for potential company products or new features for existing products, form ad hoc teams, and together build prototypes that might someday see the light. Corporate hackathons represented a mental break from everyday routine and enabled the all-important social purpose of getting to know other employees outside of your routine work. The model for exploring passion projects has become quite prevalent in the technology industry. So much so that companies can even outsource the organizing and hosting of a hackathon event to third parties [11].

Collegiate hackathons have been in existence for the past ten years. PennApps, at the University of Pennsylvania, claims to be the first and the largest student-run hackathon [6,7]. Typically, in collegiate hackathons, small teams of students, from the same institution, work together over a weekend to create solutions to some problems. Most of these events receive heavy corporate sponsorships so that participating teams do not have any out-of-pocket expenses. In return, most often, corporate sponsors dictate the kinds of problems, and the APIs that will be used at a hackathon. For corporate sponsors, these serve as marketing, recruiting, and idea-sourcing opportunities.

"Not only will we provide you with a weekend's worth of swag, meals, drinks, and snacks and a place to crash when you need a break from coding." [6]

Most collegiate hackathons are student-run affairs. In addition to hacking, organizers may run training workshops, industry and alumnae panels, etc. Hackathons tend to be enriching at several levels and go well beyond the coursework, mentoring, and research opportunities that computer science departments typically offer. However, this also raises several questions: What do students learn at these events? What are the demographics of participants in these events? Is the environment inclusive? Why do students choose (or not) to attend them? How many hackathons are too many for a student to attend in a semester/year?

"...many times the most important

Python has no types
Like no windshield, no seat belt
with deadly snake

aspect of a hackathon is the community it generates and skills that inexperienced hackers walk away with." [6]

In a way, hackathons are modern versions of various programming and robot competitions that have been in existence for much longer. Prof. Nourbakhsh and colleagues at Carnegie Mellon University have had extensive experience running and studying numerous robot competitions. They observed that student experience at robot competitions tends to be intense, short-term, and—due to its competitive setup—tends to negatively impact participant diversity [5]. Moreover, they also questioned the impact on student learning and professional development. They concluded that after the intense experience of a robot competition students essentially returned to regular coursework with little impact on their studies.

Only recently have formal studies of the social, educational, and professional impact of hackathons started to surface. It has been reported that participants at hackathons in the United States largely come from white and Asian male populations with very few women (less than 10-15%), few non-binary, and few students from underrepresented groups [3,4,7,10]. This tends to create an environment where students from non-majority groups feel marginalized and unwelcome [3,10]. Among the negative aspects of hackathons, the most prominent were novice fears, physical discomfort, too much focus on corporate-led ideas, building superficial

demos without much substance, overly competitive, and a prevalent programmer hacker culture [10].

Issues like those mentioned above were not unknown to organizers of early programming and robot competitions. For example, Nourbakhsh et al. deliberately redesigned their recent robot workshops to be free of competition and fashioned more inclusive participatory design workshops [5]. As it is in the design of curricula and courses, one must consciously design the hackathon experience to be more welcoming and inclusive. Richard et al. describe a small branch-off from the PennApps hackathon that was designed for the explicit goal of broadening participation [7]. At my own institution students run a SisterHacks hackathon—a hackathon for students from the Seven Sisters colleges: Barnard, Bryn Mawr, Mount Holyoke, Smith, Vassar, and Wellesley Colleges [8].

"Our goal is to empower women in tech by providing a supportive and enriching environment to learn, experiment, and collaborate.*

**We realize that not all students of Seven Sisters schools identify as women and we welcome all transgender, non-binary, and gender nonconforming students." [8]*

However, these represent a very small fraction of the hundreds of hackathons that are run every year in the United States and where several tens of thousands of students participate. Richard et al. mention that an explicit aspect of inclusive hack-

Advertise with ACM!

Reach the innovators
and thought leaders
working at the
cutting edge
of computing
and information
technology through
ACM's magazines,
websites
and newsletters.

Request a media kit
with specifications
and pricing:

Ilia Rodriguez
+1 212-626-0686
acmm mediasales@acm.org

OPINION

Lab Culture versus Hackathons

athon design is to create approachable themes for ideas that are appealing to a broader audience. Also, the design space itself should be designed to foster social interaction and collaboration. However, success in these events is not just about following a prescribed set of directives.

Most of these are still largely student run affairs. Faculty mentoring is an important component of success for student-led hackathons [7]. There is no doubt that co-curricular activities like hackathons can potentially play a vital role in student learning and professional growth. Yet, to implement such activities in a well-designed, engaging, and inclusive manner requires additional departmental and staffing resources [2].

Lab Culture Redux

In my mind, creating a supportive and collaborative lab culture is even more crucial in the age of hackathons. Labs represent *in situ* informal learning environments. These are venues where students have a daily existence and that can go well beyond the short, intense, and largely competitive experience of a hackathon. Besides, not all students participate in hackathons. Implications for creating (or recreating) the lab culture also begin with the choice of software, platform, and the design and structure of courses. For example, in my department, we have started teaching most of our courses, including introductory courses, on computing platforms (hardware, OS, and software) that require students to do most of their work in the computing labs. Beginning students learn to use systems and software that is not easily installed on their laptops and that brings them into the labs outside of structured lab hours.

As a department, we then must deliberately create a supportive and collaborative environment and culture. We're finding that when our students are more engaged in their everyday coursework, they're spending more time outside of classes in the department lounges and labs, they tend to socialize more with their peers, and

A simple, disconnected graph of my emotions:

they find that faculty are easily accessible just down the hallway. Occasionally, they leave us, and their peers, interesting nuggets to ponder (see pictures). Now, that is a good thing. ❖

References

1. Grijpink, F., Lau, A., and Vara, J. Demystifying the Hackathon. October 2015; <https://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/demystifying-the-hackathon>. Accessed 2019 March 5.
2. Hennessey, K. F., Parham-Mocello, J., Walker, H. Co-Curricular Activities in Computer Science Departments. Birds-of-a-feather session at ACM SIGCSE 2019.
3. Kos, B. A. The Collegiate Hackathon Experience. Extended Abstract. In *Proceedings of ICER 2018*. (ACM, New York, August 2018), 274-275.
4. Nandi, A., Mandernach, M. Hackathons as an Informal Learning Platform. In *Proceedings of the ACM SIGCSE 2016*. (ACM, New York, March 2016), 346-351.
5. Nourbakhsh, I. Robot Diaries: Creative Technology Fluency for Middle School Girls, In *IEEE Robotics and Autonomous Systems*, 2009.
6. PennApps; <http://2019w.pennapps.com/#about>. Accessed 2019 March 5.
7. Richard, G. T., Kafai, Y. B., Adleberg, B., Telhan, O. StitchFest: Diversifying a College Hackathon to Broaden Participation and Perceptions in Computing. In *Proceedings of the ACM SIGCSE 2015*. (ACM, New York, March 2015), 114-119.
8. SisterHacks; <http://sisterhacks.co/>. Accessed 2019 March 5.
9. Toma, L. and Vahrenhold, J. Self-Efficacy, Cognitive Load, and Emotional Reactions in Collaborative Algorithms Labs - A Case Study. In *Proceedings of ICER 2018*, (ACM, New York, August 2018).
10. Warner, J., Guo, P. J., Hack.edu: Examining How College Hackathons Are Perceived by Student Attendees and Non-Attendees. In *Proceedings of ICER 2017*, (ACM, New York, August 2017).
11. Your Ideas Are Terrible; yourideasareterrible.com/the-ultimate-guide-to-corporate-hackathons/. Accessed 2019 March 5.

Deepak Kumar
Computer Science
Bryn Mawr College
Bryn Mawr, Pennsylvania 19010 USA
dkumar@brynmawr.edu

DOI: 10.1145/3322639

Copyright held by author/owner