Engagement in Practice: Teaching Introductory Computer Programming at County Jails

Dr. Theresa Anne Migler-VonDollen, California Polytechnic State University, San Luis Obispo

Theresa Migler-VonDollen currently lectures in the Computer Science department at Cal Poly San Luis Obispo. Previously, she taught in the Computer Science department at UC Santa Cruz. Theresa earned a PhD in Computer Science at Oregon State University in 2014. She also holds MS and BS degrees in Mathematics from Cal Poly San Luis Obispo.

Theresa is a highly effective educator and a recent nominee for "most supportive professor" at Cal Poly. She advises MS and senior project students at Cal Poly and actively supports several student clubs including: Women in Software and Hardware (by attending the Grace Hopper Conference) and the Indian Student Association (by serving as the faculty advisor). While at UC Santa Cruz, Theresa developed an innovative course in social networks, which served as an engaging introduction to graph theory and game theory concepts for students from across the university.

Theresa's research interests center around computer science theory and algorithms with emphasis on applications throughout computer science and across disciplines. Specific areas of interest include: graph theoretic algorithms, approximation algorithms, streaming algorithms, and graph visualization. She also devotes research time to the study of computer science education in underrepresented and low income populations. One of Theresa's current projects involves teaching programming and computational thinking at jails in SLO county; a project in which she has involved several Cal Poly Computer Science students.

Dr. Lizabeth T Schlemer, California Polytechnic State University, San Luis Obispo

Lizabeth is a Associate Dean at Cal Poly, SLO. She has been teaching for 22 years and has continued to develop innovative pedagogy such as project based, flipped classroom and competency grading. Through the SUSTAIN SLO learning initiative she and her colleagues have been active researching in transformation in higher education.

Engagement in Practice: Teaching Introductory Computer Programming at County Jails

Abstract

We design an introductory course in computer programming and successfully deliver the course to two local jail populations. We discuss the structure of our program and the adaptation of traditional computer science teaching methods to the jail setting. We identify effective instructional approaches to address the unique challenges faced by in-custody students. We discuss the program's inclusion of undergraduate students and we explore assessment and future directions.

Our evaluation of the program centers on the following defined goals for in-custody students: to be empowered in logical thinking and mathematical skills, to elevate technical literacy, to create a project to take home upon release, to be prepared for success at an introductory computer programming course at our local community colleges upon their release. For undergraduate teaching assistants, we state the following expected outcomes: to have an expanded world-view and to improve their teaching and communication skills.

We evaluate impact and outcomes of our program using interviews and surveys conducted upon completion of the course. Over half of students who complete the course express an intention to continue computer science education at a local community college. We found that many students enjoyed creating programs and were proud of their success in creating these programs. Based on written reflections. Many of our undergraduate student assistants state that they learned a tremendous amount from this experience. We also observe improved teaching and communication skills.

1 Program Overview

Our four-week introductory computer programming course follows a university-style schedule: two ninety minute lecture periods per week along with a separate weekly two hour lab session. Lecturers from California Polytechnic State University, San Luis Obispo (Cal Poly), serve both as course designers and as head instructors. Undergraduate students also from Cal Poly assist with instruction and lab exercises. During each lecture period, at least one head instructor is present. Jail guards escort students to and from class, but are otherwise not present. All instructors and assistants are volunteers.

We deliver the course inside the jail facility, in a classroom equipped with a television screen / projector and 7 laptop computers (with internet disabled). The classroom is centrally located, serving both the male and female jails (although not at the same time).

Using the Python programming language for all instruction, we cover the following programming concepts: basic arithmetic, built-in Python functions (e.g. "print", "type"), user input parsing, simple data types, user-defined functions, parameters, simple programs, boolean logic, and loops.

We identify and evaluate the following objectives for in-custody students: to be empowered in logical thinking and mathematical skills, to elevate technical literacy, to be prepared for success at an introductory computer programming course at our local community colleges upon their release. We also identify the following goals for Cal Poly teaching assistants: to have an expanded world-view and to improve their teaching and communication skills.

To date, we have delivered our introductory computer programming course four times, each time to a different incarcerated population: women's jail in the summer of 2016, men's jail in the fall of 2016, women's jail in the fall of 2017, and the men's jail in the winter of 2018.

The remainder of this paper is organized as follows: first, we discuss our in-custody-student population, followed by a description of the jail teaching environment. Next we describe the structure of our program, the materials that we cover, and the inclusion of undergraduates in delivery of the course course. Finally, we cover assessment against expected outcomes as well as future directions. From this point forward, we refer to students-in-custody simply as "students" and our Cal Poly undergraduate teaching assistants as "teaching assistants".

2 Related Work

Two documented successful large-scale programs exist to teach programming and other computer skills to inmates. The most famous of these is The Last Mile at San Quentin State Prison, which enrolls inmates in a six-month, intensive computer programming class that covers web development using Javascript, CSS, and HTML. Participants in this program meet four days a week, eight hours a day. This program is modeled after a Hack Reactor bootcamp [1]. Another large-scale effort at Folsom Prison, offers female inmates the opportunity to take courses in Computer-Aided Design [2].

To our knowledge, the program we describe in this paper represents the first documented, small-scale, computer programing course at county jails that involves undergraduate students as instructors.

3 Students

There are no prerequisites for the course, it is offered to any interested person in custody. As such, the student population has a diverse educational background.

Although every effort is made to ensure that the students that begin the course will remain eligible to attend until the conclusion, the student population is in a state of constant flux. A student who is not yet sentenced may be eligible to begin the class, but may be forced to miss class sessions for court dates. Students are sometimes moved to a different population within the jail, to state prison, or may be released early. The in-flux nature of the student population requires instructional design based on small, self-contained lessons. To maximize the program's impact,

each lesson and corresponding lab activity must by valuable on its own, in the case that a student is unable to complete the course.

Students also face personal challenges. Many students deal with withdrawals from drug dependencies or uncertain family situations. Some have anxiety about their current situation or their future in jail and eventual release. These factors, too, drive instructional design choices. We find that forward-looking or family-focused exercises are most engaging. Examples include developing an application to calculate anticipated release dates or simple games or word-substitution programs that could be demonstrated to students' children.

Despite the many challenges, we have noted that the students are kind, grateful, generous with their fellow students and themselves, graciously funny, and hard working. They give us hope for the future, which is why we continue to go to jail and work for expanding their horizons.

4 The Environment

After arriving at the jail, volunteers go through a brief, 10-minute check in process. Instructors and assistants must follow a strict dress code and adhere to all rules in the jail. Students are permitted a pencil, printed course materials, and a folder. Guards escort students to class and after an hour and a half, guards return to escort students back to their dormitories.

Anyone who assists with the program must attend a jail orientation class offered by our partner agency, Restorative Partners, and the jail. During the session volunteers are taught how to behave in jail, risks of being in jail, and how to minimize these risks. Once a volunteer has completed training they are required to obtain security clearance, which involves fingerprinting and standard paperwork.

5 Objectives & Course Material

Our course goals are:

- To empower the students.
- To elevate the technical literacy of the students.
- To help the students gain confidence in their logical thinking and mathematical knowledge.
- To give the students a project to take home upon release to demonstrate to their family and friends.
- To prepare students for an introductory computer programming course at our local community colleges.

With these goals in mind, we carefully prepare our course. The course aims to be welcoming and encouraging while also rigorous and logically demanding.

We choose Python as the programming language for the course due to its approachability and wide usage. With the compressed, four week schedule and widely varying student skill level, Python's simple yet powerful syntax avoids a steep learning curve [3]. Furthermore, the

specific community colleges for which we are indirectly preparing our students use Python in their introductory computer science courses.

We chose to develop our own course material for two reasons. First, to minimize project expenses, and secondly because hard cover books are not allowed in jail. We carefully prepare "offline" assignments to reinforce the programming concepts taught in class without requiring a computer. Many of our offline assignments take the form: "What do you think the output would be if _____?" See theresamigler.com/jail-classes/ website for lecture materials and offline programming assignments.

We try to motivate our programmatic ideas through interesting examples using both math and text-based operations. We also try to bring our examples to life by including students' families in our problems. For example, we introduce engaging programs that our students can take home and show off to their children and families. We find this to be important because around 75% of incarcerated women have children under the age of 18 (2.4 million children have one or both parents in a correctional facility) [4]. One ancillary gaol of this program is to help incarcerated parents to reconnect with their children in a positive way upon their release.

We have some pre-planned assignments, such as a moon weight calculator which asks for a user's weight on earth and calculates his or her weight on the moon. We also build flexibility into the curriculum, incorporating student ideas whenever possible. For example, after working on the moon weight calculator program, a student said "I wonder if we could use the same idea to come up with a sentencing calculator?". The sentencing calculator involved accepting as input (a) an initial sentence duaration, (b) eligibility for "half time" and (c) if you were good and got your 10% "kick". This program was exciting to create because we hadn't yet discussed "if" statements. We had to work around "if" statements by using zeros and ones as input:

```
def sentence():
 original = int(input("What is your sentence (in days)?  "))
 goodTime = int(input("Will you get good time?  1 for yes, 0 for no:"))
 kick = int(input("Will you get your kick?  1 for yes, 0 for no:"))
 timeToServe = original - original*goodTime*.5 - original*kick*.1
 print("You will serve ", timeToServe, "days.")
```

The brilliant program was not only hugely motivating for our students, but it also helped our teaching assistants to understand a little more about sentencing and time served.

6 Teaching Assistants

Our teaching assistants assume a large role in the course. Whenever possible they teach the material while the head instructor offers guidance only. Assistants are responsible for setting a welcoming tone for each class by choosing an "ice breaker" type of question, such as "If money were no object, what would be your dream occupation?" or "What is your favorite holiday and why?"

We recruit teaching assistants via in-class announcements about the volunteer opportunity in upper division computer science courses at Cal Poly. The students who are interested generally have some teaching/tutoring background. Teaching assistants are supported through conversations with the instructors and each other. Further, Restorative Partners offers discussion groups and get-togethers for all jail volunteers.

Our goals for our teaching assistants are to have an expanded world-view and to improve their teaching and communication skills. When asked to share their experience:

"I found the experience very rewarding and unique. The most interesting thing about working there this quarter was realizing how bidirectional the teaching was. I feel like I learned just as much from the students as they did from the class."

"While I initially had reservations about the potential success of regularly visiting a jail to teach women how to code, a skill I had previously thought of as reserved to those with the privilege of an expensive education or at the very least regular Internet access, I can happily admit that I was countlessly proven wrong in this assessment."

7 Assessment

At the outset of the course, we ask students to articulate their goals, using the prompt question: "What do you hope to gain from this class?" These student goals shape and inform our instructional approach. Sample responses include:

- "I want to know everything I can about a computer."
- "A greater knowledge of the inner workings of computer programs and hopefully a specific skill that I don't already have."

When a four-week class concludes, we ask: "What did you gain from this class?" Student responses qualitatively confirmation of stated outcomes and goals of the program:

- "I gained knowledge and an academic sense of direction for the future."
- "A great understanding of programming and how it can actually apply to something I would pursue in the future."

To drive refinement of course exercises, we also ask: "What was your favorite topic?" Emphasizing games and other approachable applications of simple computer programs, student responses include:

- "Paper rock scissors" (We ended the course with a program to "play" the paper-rock-scissors game.)
- "Creating functions."

Two main factors complicate assessment. First, a student might drop out for reasons beyond their control, preventing us from conducting a closure survey with that student. Second, due to privacy policies, there is no way to follow up with students upon their release. Because of this, we are unable to monitor if we are successful in our goal for having students enroll in community

college upon their release or have a program that they are proud to show to their families. We intend to work more closely with local community colleges to develop privacy-sensitive surveys that would allow us to gather aggregate, anonymous data to assist with full-circle assessment.

8 Future Work

Through Cal Poly, we are able to offer extended education course credit to students who complete our course, including a transcript from Cal Poly. We are working with local industry and the jail to tailor and expand our curriculum in a way that could lead more deirectly to employment. One of the most common questions that we get asked in class is: "Is computer programming a felon-friendly occupation?". The answer that we hear from industry is an overwhelming "Yes." We hope to partner with local companies so that the interested student may be mentored as they continue their computer programing education.

9 Many Thanks

We are extremely grateful for the support and guidance of Sister Theresa Harpin, Olivia Lopes, and Robyn Morton at Restorative Partners. We are thankful for Dr. Sussan Einakian and our student assistants, Yusuf Bahadur, Hayley Cushing, Cassie Hamilton, Larry Hu, Caleb Kim, Christ Moranda, Terry Wambolt for implementing this program.

10 References

[1] B. Keller, "A Prison that Offers Hope", TIME, June 30, 2016 [Online].
Available http://time.com/4389086/a-prison-that-offers-hope/?iid=sr-link2 [Accessed January 19, 2018]
[2] L. Newell, "New computer design program offers Folsom female inmates job-skill training", The Folsom Telegraph, March 12, 2014 [Online]. Available http://www.folsomtelegraph.com/article/new-computer-design-program-offers-folsom-female-inmates-job-skill-training [Accessed January 19, 2018]
[3] C. Stevenson, "The Development of a Course Covering Scripting Languages and Their Applications", Consortium for Computing Sciences in Colleges, June 2012
http://dl.acm.org/citation.cfm?id=2184451.2184484
[4] J. Smyth, "Dual Punishment: Incarcerated Mothers and Their Children", Columbia Social Work Review, p ,33, 2012. [Online]. Available https://cswr.columbia.edu/article/dual-punishment-incarcerated-mothers-and-their-children/ [Accessed January 19, 2018].
[5] C. Zoukis, "Prison Teachers: A Different Type of Educator", Huffpost, November 4, 2015 [Online]. Available https://www.huffingtonpost.com/christopher-zoukis/prison-teachers_b_8473178.html