Volume 16, No. 1
February 2018
ISSN: 1545-679X

INFORMATION SYSTEMS EDUCATION JOURNAL

In this issue:

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **ISCAP** (Information Systems and Computing Academic Professionals). Publishing frequency is six times per year. The first year of publication was 2003.

ISEDJ is published online (http://isedj.org). Our sister publication, the Proceedings of EDSIGCON (http://www.edsigcon.org) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the EDSIGCON conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is under 40%.

Information Systems Education Journal is pleased to be listed in the Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org. Special thanks to members of AITP-EDSIG who perform the editorial and review processes for ISEDJ.

INFORMATION SYSTEMS EDUCATION JOURNAL

Editors

2017 ISEDJ Editorial Board

"Hour of Code": A Case Study

Jie Du
dujie@gvsu.edu
Grand Valley State University
Allendale, MI, 49401, USA

Hayden Wimmer
hwimmer@georgiasouthern.edu
Georgia Southern University
Statesboro, GA, 30460, USA

Roy Rada
rada@umbc.edu
University of Maryland Baltimore County
Baltimore, MD, 21250, USA

Abstract

This study investigates the delivery of the "Hour of Code" tutorials to college students. The college students who participated in this study were surveyed about their opinion of the Hour of Code. First, the students' comments were discussed. Next, a content analysis of the offered tutorials highlights their reliance on visual programming in stylized languages with continual feedback in gaming contexts. Difficulties in delivery stem in part from the poor organization of tutorials from Code.org which makes it difficult to locate suitable tutorials. Based on the analysis of the students' comments and the content analysis of the "Hour of Code" tutorials, the authors suggest that a deeper alignment of marketing, teaching organizations, and content providers would help sustain the type of initiative exemplified by the Hour of Code.

Keywords: Advocacy, Hour of Code, code.org, online tutorials, introductory computer programming, survey, literature review.

1. INTRODUCTION

Much has been written about efforts to spread computer science education. The difficulties that students face in learning to program have been studied by Kinnunen and Simon (2012). Difficulties students encounter include required for systems development such as problem-solving, coding, and testing (Scott, 2008). A National Science Foundation sponsored study concluded that further training of computer science teachers was crucial (Goode & Margolis, 2011). The Berkeley Foundation for Opportunities in Information Technology concluded that helping underserved students appreciate computing has acquired predictable, year-round funding (Crutchfield et al., 2011). Such conclusions about historical continuity, personnel training, and funding have been re-discovered time and again as crucial to education. Many other pre-conditions for a successful education initiative could be noted.

Code.org is a non-profit organization that is dedicated to bringing computer programming into the mainstream dialogue and raising national awareness about this issue. As part of the initiative, Code.org hosted an Hour of Code curriculum of 100+, one-hour-long, computer science activities. The term "Hour of Code" may also refer to a specific, one-hour introduction to computer science that is organized by Code.org.

The "Hour of Code" activities are game-based. Students can learn computer science basics by playing a game, such as Minecraft. The "Hour of Code" tutorials teach students how to utilize problem-solving skills and logic to win the games. The "Hour of Code" tutorials are on-line, web-based, and work on computers or mobile devices. The "Hour of Code" tutorials are designed for all ages and are available in over 45 languages.

A computer science education nurtures problem-solving skills and creativity. The goal of the Hour of Code is to demystify code and show that anybody can learn the basics. Participants in the Hour of Code will hopefully learn that computer science is fun and creative. Since Code.org was launched in 2013, over 100 million students have participated in 200,000 code.org events worldwide.

How successful was the Hour of Code and what can be learned by studying what transpired? To address this question, this paper presents one study on college students to investigate how to facilitate the delivery of an Hour of Code to students. An analysis of the offered tutorials highlighted their reliance on visual programming in stylized languages with continual feedback in gaming contexts. Among other difficulties, the organization of tutorials from Code.org made it difficult to determine what tutorials to use.

The remainder of this paper is organized as follows. Section 2 reviews the published documents about the "Hour of Code" initiative. Section 3 presents a survey study on the delivery of the Hour of Code. The challenges of improving the Hour of Code are discussed in Section 4. Section 5 concludes the paper.

2. BACKGROUND

Scholarly Publications

The authors queried the ACM Digital Library, IEEE Xplore, and Google Scholar for the string "code.org" on Nov. 1, 2013. Many retrieved citations were false positives or redundancies. For instance, in the false positive category "code.org" occurred in pharmacy articles and in the redundancy category several citations were to the same interview with the founder of Code.org (Hadi Partovi). The query on the ACM Digital Library for "Code.org" on Nov. 1, 2013 retrieved 11 citations which after excluding false positives and redundancies reduced to three:

- One citation (Ardis & Henderson, 2013) referenced recent efforts to encourage coding, including quoting President Barack Obama and Code.org but then argued that such a focus on coding was counter-productive because the better effort was to teach students the principles of computer science.
- One citation was to a multi-page interview with the founder Hadi Partovi of Code.org (Snyder, 2013). Partovi emphasized his enthusiasm for the code.org vision but also that he was the only employee of Code.org and needed more time to sleep.
- Only one article (Lee, Ko, & Kwan, 2013) was a research paper but that paper only says about Code.org: "Recent press about Code.org and other efforts to increase computing literacy have begun to attract millions of people to learn computer programming."

The query on IEEE Explore returned two citations of which one was a false positive and the other was an editorial about the future of Code.org which spoke in glowing terms of the endless possibilities (Wilson, 2013).

The query on Google Scholar for "code.org" retrieved 674 citations. The first 50 citations had nothing about Code.org as related to computer programming, and most citations referred to a pharmaceutical agent. The query was modified to include the term "Partovi". Then the query retrieved exactly 6 citations. Only one of those was relevant and not already covered by the ACM and IEEE searches, and it was an online report about attracting women into computing (Mueller, 2013) and referred to Code.org as an example of an advocacy campaign for getting school-aged children interested in coding.

Library	Number of Citations		
	Retrieved	Unique and Relevant	About Advocacy
ACM Digital Library	11	3	3
IEEE Xplore	2	1	1
Google Scholar	6	1	1
Totals	**19**	**5**	**5**

Table 1. This table shows the number of citations retrieved from each library by category in 2013.

Across three retrieval engines, queries returned only a handful of relevant articles (see Table 1).

More specifically, a total of 19 citations were retrieved. Five of those were unique and relevant, and all 5 were about Code.org's advocacy situation in 2013.

Another Look at Publications

The authors queried the ACM Digital Library, IEEE Xplore, and Google Scholar for the string "Code.org" again on Jan 13, 2014 and saw a doubling of the number of citations between November 2013 and January 2014 thereby demonstrating the growing interest in Code.org. A query on the ACM Digital Library for "Code.org" on Jan 13, 2014 retrieved 22 citations which after excluding false positives and redundancies included, of course, the 3 retrieved on Nov. 1, 2013 but also 6 more:

- Three Communications of the ACM editorials or business reports, one each in the Nov 2013, Dec. 2013, and Jan 2014 issues.
- Two SIGCSE Bulletin news articles in the October 2013 issue.
- An Inroads Dec 2013 opinion piece.

ACM has aligned itself with Code.org and the increasing number of publications supports the advocacy activity of Code.org. The query on IEEE Explore returned only one further citation which was an announcement that the IEEE Computer Society was a promotional sponsor of the hour-of-code (IEEE Computer, 2013). A query on Google Scholar for "Code.org Partovi" returned only two new, relevant citations: a news story on a web site and a bachelor's thesis from a Dutch university that pointed to Code.org as an example of an advocacy effort for computer education (Verkroost, 2013).

In order to dig deeper into the news story aspect, LexisNexis was accessed. A query for 'Code.org' on LexisNexis on Jan 14, 2014 returned 1,000 citations. The first 50 showed that many newspaper articles accompanied Computer Science Education Week and gave credit to Code.org. Code.org gave $10,000 grants to schools for the purchase of laptops with some grants announced during Computer Science Education Week. Extracts from a newspaper article in the Charleston West Virginia Gazette illustrated the kind of publicity generated (Charleston Gazette, 2013): "... all of the 670 students at Winfield Middle School will take part in the largest education event in history: The Hour of Code. .... To aid them in this endeavor, Code.org has awarded the school a $10,000 grant to purchase laptop computers. ... " For $10,000 to generate this much publicity is impressive. Several California newspapers noted that the Hour of Code was a publicity stunt but that a publicity stunt was needed (Cassidy, 2013): "Yes, we can all agree that this week's big Hour of Code initiative is a publicity stunt, ... A publicity stunt is exactly what we need." A few news reports showed longer-lasting, larger-scale results, such as an announcement from Chicago. Chicago announced on Dec. 10, 2013 a new initiative for Chicago Public Schools (US Official News, 2013): "In the next three years, every high school will offer a foundational 'Exploring Computer Science' course. ... Chicago Public Schools will receive free computer science curriculum and ongoing professional development and stipends for teachers to implement this plan thanks to a district partnership with Code.org, a nonprofit dedicated to increasing access to a computer science education."

The first 50 results from LexisNexis showed this pattern:

1. Eight news stories explored issues addressed by Code.org's hour-of-code,
2. Five news stories were about local schools participating in the hour-of-code (without mention of a $10,000 grant).
3. Four news articles (about different school districts in different states) noted a $10,000 grant from Code.org and the hour-of-code initiative.
4. Four news stories were from outside the US, such as one from Sri Lanka about a software company that helped several children experience a Code.org tutorial (Daily Mirror, 2013).
5. Three were official announcements, one from the City of Chicago, one from Microsoft, and one from the Patent and Trademark Office (which noted that Code.org trademarked 'CODE').

The remaining 26 citations were either irrelevant or redundant.

Publications in More Recent Years

The authors queried the ACM Digital Library for the string "Code.org" again on June 6, 2017. The query on the ACM Digital Library for "Code.org" retrieved 45 citations which after excluding false positives and redundancies from previous searches reduced to 13:

- One Communications of the ACM editorial in the Feb. 2017 issue.
- One SIGCSE Bulletin news article in the January 2015 issue.
- Seven ACM Inroads opinion pieces from Dec. 2014 to Nov. 2016.

- Two abstracts from ACM technical symposium on Computer science education proceedings. Hanley (2016) discussed using Myna, a Programming by Voice tool to support children with a mobility disability to experience Code.org's educational opportunities. Meeker (2014) dsicussed the need to better promoted the "Hour of Code" initiative in local schools.
- Two research papers discussed how the hour of code can be utilized to improve the learning process. Piech, et al. (2015) developed a family of algorithms that can predict the way an expert teacher would encourage a student to make forward progress and then used the algorithms to automatically generate hints for the Hour of Code. Theodoropoulos et al. (2016) assessed the learning effectiveness and motivational appeal of the Code.org's activity named K-8 Intro to Computer Science. Seventy-seven students of two Greek high schools participated their study and the results show that these specific educational computer games provide a high-quality learning experience.

Across these document databases, one sees that Code.org received substantial publicity for its "Hour of Code" initiative. News stories corroborate the initiatives of Code.org which are to train teachers and to encourage states to consider computer science education mainstream.

3. UNIVERSITY CLASSROOM SURVEY

A study was conducted to investigate how educators could deliver the Hour of Code to students. The purpose of this study is to 1) examine the effect of the Hour of Code on students' attitude toward programming, and 2) gain insights on improving the "Hour of Code" initiatives to better promote computer science education.

Lesson Plan

One of the authors has been teaching introductory computing courses at one Midwestern public, master's granting university for years. Since summer 2014, the author started introducing the "Hour of Code" tutorials in the introductory computing course. The author usually started the class with a 6-min video that begins with this quote from Steve Jobs "Everybody in this country should learn how to program a computer because it teaches you how to think" and follows with multiple short interviews where people discussed the role of computers in their life and how important they believed it would be that everyone learn to code for, at least, an hour. The interviewees represented a broad spectrum of types of people well-known to Americans and included Bill Gates, the founder of Microsoft, and Chris Bosh, a famous American basketball player. Next, students were asked to undertake the tutorial "Write Your First Computer Program" from the category of "Tutorial for Beginners" at code.org. Students were asked to take a pre- and post-survey about their experience at Code.org.

Data Collection

This practice was first implemented in 2014 and then has been repeated every year since then. Thus, data was collected from 2014 to 2017 from the students enrolled in an introductory computing course at one Midwestern public, master's granting university. As a result, the data set yielded 255 usable responses (116 in 2014, 45 in 2015, 47 in 2016, and 47 in 2017). Most of the participants are freshmen and fresh-women with a spectrum of majors including business, accounting, criminal justice, allied health sciences, geography, hospitality tourism management, and psychology.

Both qualitative and quantitative data were collected. The quantitative data were collected using Likert-Scale survey questions. The quantitative data analysis from the 2014 data set shows that the Hour of Code has a positive impact on students' attitude toward programming. The detailed survey questions and results from the 2014 data set can be referenced at (Du, Wimmer, & Rada, 2016).

For qualitative data collection and the focus of this paper, participants were asked to provide additional comments regarding programming as well as the "Hour of Code" tutorials at the end of each survey. As a result, 39% of the participants provided additional comments. That is, 97 qualitative comments (55 from the pre-survey and 42 from the post-survey) were collected.

Data analysis

The authors analyzed the students' feedback on their experience at Code.org. Students' feedback on their attitude toward programming was first presented. A discussion on challenges of delivering the hour of code follows.

Attitude toward Programming

Students' comments in the pre-survey before they completed the "Hour of Code" tutorial are incredibly negative (see Figure 1). The top 3 high-frequency quotes show the participants' frustration:

- *I have no clue what I am doing. (25%)*
- *No previous programming experience. (23%)*
- *I am confused. (16%)*

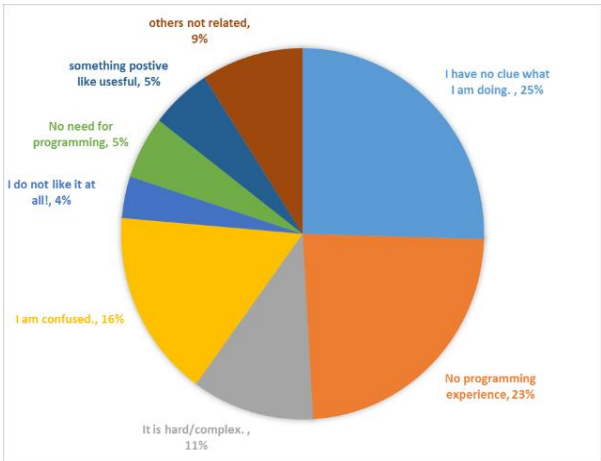

Figure 1 Participants' comments in the pre-survey

It is interesting to find that participants showed more interests on programming after completing the "Hour of Code" tutorial at Code.org (see Figure 2). Although some comments address the difficulty of learning to program, most of the comments (nearly 70%) are very positive. The selected quotes from students' comments in the post-survey illustrate that students appreciated the "Hour of Code" tutorial:

- *I have no realized programming is not just a bunch of symbols on a computer, it is a way of thinking, processing, and problem solving*
- *I think it is interesting.*
- *That was a very cool and helpful assignment. I have never done anything like that before.*
- *I am interested in coding quite a bit, it can unlock a lot of potential!*
- *Programming seems a lot easier.*
- *The tutorial put it into terms that were easier to understand.*
- *This seemed to simplify programming for me.*
- *I may look into programming more.*
- *Learned a bit more about coding.*
- *Interesting tutorial, enjoyed programming.*
- *Programming seems very helpful for many occupations.*
- *I have been playing Minecraft for many years but not like this! Very cool!*

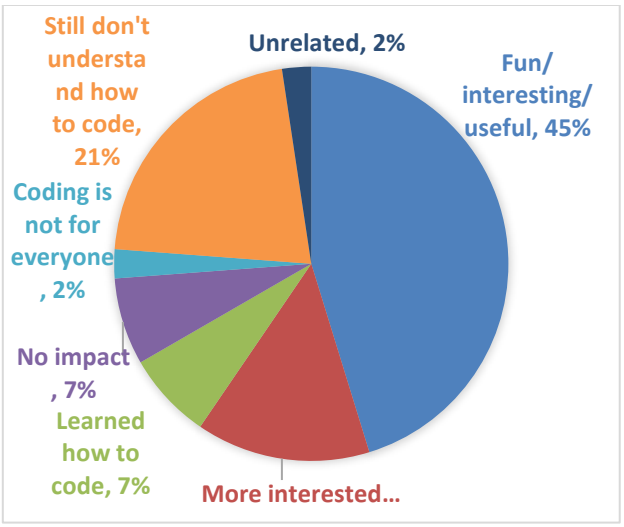

Figure 2 Participants' comments in the post-survey

Observations of Delivering the Hour of Code

One challenge the authors faced has been accommodating the widely different skill levels of students in the class. Some students had very limited knowledge about coding, while other students already took a computing course in high schools. The authors appreciate the difficulty to identify an appropriate "Hour of Code" tutorial for the college students with widely different coding skills. The selected quotes coming from two ends of this spectrum highlight this challenge:

- *It is cool and fairly easy.*
- *I already am familiar with coding languages, not much impact.*
- *It is complicated.*
- *Very interesting, but I still feel like I don't understand very much about programming code.*

Another observation from the authors is that compared to their male peers, female students are more intimated by coding and reluctant to try it at the beginning. This might be supported by the fact that the gender gap in computing is getting worse nowadays. If nothing changes the way that we teach computing to girls, the gender gap is expected to hurt U.S. economy seriously. Schools and colleges need to initiate outreach efforts to have more women engaging with computer science at the same rate as men. One student's feedback echoed this effort: "*I think it is great to encourage girls to become interested in coding.*" Gender differences will be analyzed and presented in future work.

The authors also identified several things that Code.org might improve in the future. While

focused on K-12 education, Code.org claims that their curriculum and tutorial materials can be used for any non-commercial, computer science educational purposes, such as introductory course in colleges. There are no appropriate tutorials designed for college students as of yet on Code.org. Some participants' feedback pointed out the lack of tutorials for college students.

- *I did relatively enjoy code.org, yet it was a little childish.*
- *It is a good tutorial for children in the aspect of coding. For young adults and college students, I think it would be more beneficial if they learned a starter language like Java.*

4. CHALLENGES OF IMPROVING THE HOUR OF CODE

In this section, a content analysis on the "Hour of Code" tutorials is presented and challenges on improving the "Hour of Code" initiatives are discussed.

Organization of Tutorials

The authors analyzed the content evolution on code.org since 2014. On Jan. 13, 2014, the top level of the organization of tutorials was 7 revolving menus that categorize tutorials as:

1. for beginners
2. learn JavaScript
3. no device
4. apps for a device
5. other programming languages
6. make your own apps
7. other learning options

Within each category, each tutorial was presented with a handful of attributes to include most prominently the author, the title, a paragraph blurb, the target ages, the number of participants, and the URL. Code.org authored the tutorial called "write your first computer program" which is the famous "Angry Birds" tutorial and which had the most participants at 9,900,000.

The catalog listed 30 tutorials, although some tutorials occurred in two categories (see Appendix). For instance, MIT's "App Inventor" was listed under both the "for beginners" and the "make your own apps" categories. Tynker's "build your own game" was listed under "for beginners" but also under "other learning options". In taxonomy construction, one should define each tutorial and category by attribute values and enter a tutorial under a category when it inherited attribute values from its parent. Such a taxonomy policy was not adequately followed in the design of menus for these tutorials.

The number and the organization of the "Hour of Code" tutorials have dramatically changed since 2014. On June 6, 2017, the catalog listed 149 tutorials which can be further filtered by:

- Grades
  - Pre-readers
  - Grades 2-5
  - Grades 6-8
  - Grades 9+
- Educator experience
  - Beginner
  - Comfortable
- Student experience
  - Beginner
  - Comfortable
- Classroom technology
  - Computers
  - Android
  - iPad/iPhone
  - Poor or no internet
  - No computers or devices
- Topics
  - Science
  - Math
  - Social Studies
  - Language Arts
  - Art, Media, and Music
  - Computer Science only
- Activity type
  - Self-led tutorials
  - Lesson plan
- Length
  - One hour
  - One hour with follow on
  - A few hours
- Language
  - Blocks
  - Typing

Although the new organization incorporates more factors to facilitate locating a tutorial, a teacher might still struggle to identify which tutorials would be appropriate for which students. For instance, it is unclear how the beginner and comfortable levels were defined. Some educators might have a hard time locating appropriate tutorials due to lack of understanding about the organization and how it is defined.

Challenges

The Code.org organizers claimed that the Hour of Code was enormously successful. An email sent on Dec. 23, 2013 from Hadi Partovi (the creator of Code.org) to his supporters began with "Thank you for supporting computer science and making

the Hour of Code an amazing success. In 2 weeks, 20 million students worldwide learned an Hour of Code, including 1 in 4 US students, and half were girls!" However, neither the email nor the web site explains how the number of participants was determined. One does not need to log into a tutorial in order to use it. The authors visited the "Angry Birds" tutorial many times over different days from different computers and different countries. The server would have considered those visits to come from different participants but that overestimates the number of unique participants. The Dec. 23rd email concludes by encouraging readers to ask Congress to support the Computer Science Education Act, to ask their local schools to teach computer science, and to donate money or time to Code.org.

Many citizens of the world see the benefits that could accrue from a wider dissemination of computer knowledge. Code.org's goals would ultimately be best achieved by the government, as Code.org admits. Computer science education reaches a larger fraction of the population in many countries than it does in the US. For instance, the Kingdom of Saudi Arabia has nationalized plans that put computer labs in every secondary school and trained 6,000 teachers of computer science (Ministry of Education, 2013). In Singapore, the Ministry of Education has a country-wide, detailed master that provides a student-to-computer ratio of 2:1 in all public schools and a curriculum that is extensively supported by information technology and teaches information technology (Committee on Compulsory Education, 2000). The management of K-12 education in the US does not seem to support such centralized, country-wide results.

Implications and Limitations

This study investigates the delivery of the Hour of Code tutorials to college students. Students are not expected to become an expert of computer science in one hour. The Hour of Code is only the first step for students to learn that computer science is fun and creative. The findings of this study show that the participants became much more interested in programming after they tried the tutorials and expressed an interest to know more about coding. Educators can gain confidence after learning the "Hour of Code" tutorials that they can teach computer science even though they may not have a computer science degree. It is important to learn that everyone can learn computer science.

Nowadays computers are used in almost every industry. According to National Center for Education Statistics, there were about 60,000 computer science graduates from US institutions in 2015 and there were about 530,000 computing job opening at that year. It is estimated that by 2020 there will be 1.4 million computing jobs available while the number of computer science graduates will only be 400,000. The gender gap makes things worse. According to Accenture and Girls Who Code, 24 percent of computing jobs are held by women and this number will fall to 22% by 2025 if no changes are made on how computer science is taught to girls. It is vital for educators to 1) help students understand the importance of computer science and technology to their lives and their future career paths, and 2) teach students computer science in an easy to understand way to help them gain problem solving skills as an essential skill for life. Educators could explore the various "Hour of Code" activities and may find ones that fit in their curriculum and meet their students' need.

There are some limitations worth noting with this study. This study was limited to students at a single university. The number of survey questions are limited. Given that, various interpretations on the findings could be argued. A more diverse sample will increase the validity. More questions in the survey will help better understand participants' attitudes toward programming and their computer science skills. Gender differences on learning coding will be analyzed in future work.

5. CONCLUSIONS

The Hour of Code is claimed by its organizers to be a success. A review of the literature shows that much of what has been written about Code.org identifies it as an advocacy organization. A survey study of college students learning the Hour of Code shows that the Hour of Code stimulated students' curiosity and opened their mind to programming.

The initiative to provide an Hour of Code could be improved, and one improvement would be to better coordinate the advocacy and the tutorial delivery activities. The authors attempted to work with local colleges to institute an "Hour of Code". However, the instructors needed to reduce the scope of their plans, in part because of the difficulty of identifying the appropriate tutorial to use with their students. A better catalogue of tutorials and semi-automated aids to match a student to the appropriate tutorials might help.

While, as with any qualitative study, different researchers may come to different conclusions,

we aim to illustrate our experiences and entice other instructors to also consider Hour of Code in their classrooms. Educators can host an Hour of Code all year round. A good time to host an Hour of Code is the annual Computer Science Education Week (such as the one December 4-10, 2017).

6. REFERENCES

Ardis, M. A., & Henderson, P. B. (2013). Software Engineering != Coding. *ACM SIGSOFT Software Engineering Notes, 38*(3), 5.

Cassidy, M. (2013, December 12). Code.org Hour of Code builds a deeper understanding of the power of computing, *Contra Costa Times*.

Charleston Gazette. (2013, December 18). Winfield Middle obtains grant to teach coding, *Charleston Gazette (West Virginia)*.

Committee on Compulsory Education. (2000). *Report of the Committee on Compulsory Education in Singapore*. Singapore: Ministry of Education. Retrieved from http://www.moe.gov.sg/initiatives/compulsory-education/files/ce-report.pdf.

Crutchfield, O. S. L., Harrison, C. D., Haas, G., Garcia, D. D., Humphreys, S. M., Lewis, C. M., & Khooshabeh, P. (2011). Berkeley Foundation for Opportunities in Information Technology: A Decade of Broadening Participation. *ACM Transactions on Computing Education, 11*(3), 1-24.

Daily Mirror. (2013, December 24, 2013). Brandix i3 supports global software coding initiative, *Daily Mirror of Sri Lanka*.

Du, J., Wimmer, H., & Rada, R. (2016). "Hour of Code": Can it change students' attitudes toward programming? *Journal of Information Technology Education: Innovations in Practice, 15*, 52-73.

Goode, J., & Margolis, J. (2011). Exploring Computer Science: A Case Study of School Reform. *ACM Transactions on Computing Education, 11*(2), 1-16.

Hanley, C. R. (2016). *Programming by Voice to Support Hour of Code for Children with Motor Disabilities (Abstract Only)*. Paper presented at the Proceedings of the 47th ACM Technical Symposium on Computing Science Education, Memphis, Tennessee, USA.

IEEE Computer. (2013). Hour of Code Kicks Off to Introduce K-12 Students to Computer Programming. *Computer, 46*(11), 99.

Kinnunen, P., & Simon, B. (2012). My program is ok – am I? Computing freshmen's experiences of doing programming assignments. *Computer Science Education, 22*(1), 1-28.

Lee, M. J., Ko, A. J., & Kwan, I. (2013, August 12-14, 2013). *In-Game Assessments Increase Novice Programmers' Engagement and Level Completion Speed.* Paper presented at the ICER'13 Ninth Annual International ACM Conference on International Computing Education Research San Diego, California.

Meeker, P. H. (2014). *Inspiring a love of computer science through the education of our youth (abstract only)*. Paper presented at the Proceedings of the 45th ACM technical symposium on Computer science education, Atlanta, Georgia, USA.

Ministry of Education. (2013). IT Department Retrieved January 14, 2014, 2014, from http://www2.moe.gov.sa/english/Pages/it_department.htm

Mueller, R. (2013, March 21-22, 2013). *Attracting Females into ICT in Canada* Paper presented at the Information and Communications Technology (ICT) Talent Workshop, Ottawa, Canada.

Piech, C., Sahami, M., Huang, J., & Guibas, L. (2015). *Autonomously Generating Hints by Inferring Problem Solving Policies*. Paper presented at the Proceedings of the Second (2015) ACM Conference on Learning @ Scale, Vancouver, BC, Canada.

Scott, E. (2008). From Requirements to Code: Issues and Learning in IS Students' Systems Development Projects. *Journal of Information Technology Education: Innovations in Practice, 7*, 1-13.

Snyder, L. (2013). An Interview with Hadi Partovi. *Communications of the ACM, 56*(9), 41-45.

Theodoropoulos, A., Antoniou, A., & Lepouras, G. (2016). How Do Different Cognitive Styles Affect Learning Programming? Insights from a Game-Based Approach in Greek Schools. *Trans. Comput. Educ., 17*(1), 1-25. doi: 10.1145/2940330

US Official News. (2013, December 10). Illinois: Mayor Emanuel And CPS CEO Barbara Byrd-Bennett Announce Comprehensive K-12 Computer Science Program For CPS Students, *US Official News*. Retrieved from http://www.cityofchicago.org/

Verkroost, Y. (2013). *Seriousify and prettify our educational system! ABOUT THE USE OF (SERIOUS) GAMES AND AR IN EDUCATION*. Bachelor Project VU University Amsterdam. http://www.yordiverkroost.nl/vu/paper.pdf, Amsterdam.

Wilson, C. (2013). What's up next for Code.org. *IEEE Computer, 46*(8), 95-97.

Appendix

This appendix shows the complete taxonomy of the tutorials at http:code.org/learn retrieved by the authors on Jan. 13, 2014. The 7 left-most list items are the categories. The numbered sub-lists give the author of the tutorial followed in quotes by the title of the tutorial.

For Beginners
1. code.org "write your first computer program"
2. Scratch "create a holiday card"
3. tynker "build your own game"
4. lightbot "lightbot"
5. MIT Center for Mobile Learning "AppInventor Hour of Code"

JavaScript
1. CodeCombat "CodeCombat"
2. KhanAcademy "Introduction to Javascript"
3. codeacademy "codeacademy"
4. codeHS "Learn to Code with Karel the Dog"
5. code avengers "Learn to Code a javascript quiz game"

No Device
1. Thinkersmith "My Robotic Friends"
2. Google Education "Blockly"
3. Thinkersmith "Binary Baubles"
4. Kodable "fuzzFamily Frenzy"

Tutorial Apps
1. Lightbot "Lightbot"
2. Kodable "Kodable"
3. Hopscotch "Code on your Ipad"

Other Programming Languages
1. Grok Learning "A Taste of Python Programming"
2. Processing "Drawing with Code"
3. Robomind Academy "Program a virtual robot"
4. MakeGameswithus "Build an iPhone game in your browser!"

Make Your Own Apps
1. MakesGameswithus "Build an iPhone game in your browser"
2. MIT Center for Mobile Learning "AppInventor Hour of Code"
3. Microsoft Research "TouchDevelop"

Other Learning Options
1. Tynker "Build your own Game"
2. Microsoft Research "Touch Develop"
3. University of Colorado "Make a 3D Frogger Game in an hour"
4. Alice Project "Intro to Programming with Alice 2"
5. RunRev "Everyone can code with LiveCode"
6. Washington University Computer Science "Looking Glass"