

Developing Inclusive Outreach Activities for Students with Visual Impairments

Stephanie Ludi

Rochester Institute of Technology
134 Lomb Memorial Drive
Rochester, NY 14613
+1.585.475.7407
salvse@rit.edu

Thomas Reichlmayr

Rochester Institute of Technology
134 Lomb Memorial Drive
Rochester, NY 14613
+1.585.475.2852
tjrese@rit.edu

ABSTRACT

Despite advances in assistive technology, relatively few visually impaired students participate in university-level computing courses. Significant factors in this under representation include lack of relevant precollege preparation, lack of role models, access to resources, and the highly visual nature of modern computing. This paper describes the development of inclusive activities and materials for use in a summer workshop for precollege students with visual impairments. All activities utilized commercial technologies in the areas of robotics and programming using Lego Mindstorms NXT. The workshop activities are designed to provide a foundation in computing that encourages students to pursue computing in high school and beyond. In addition to activity design, initial results and lessons learned from the summer workshop will be presented.

Categories and Subject Descriptors

K.3.1 [Computers and Education]: Computer Uses in Education –*collaborative learning*, K.3.2 [Computers and Education]: Computer and Information Science Education –*computer science education*, K.4.2 [Computers and Education]: Social Issues – *assistive technologies for persons with disabilities*.

General Terms

Design, Experimentation, Human Factors,

Keywords

Outreach, inclusion, accessibility

1. INTRODUCTION

Like other underrepresented groups, students who are visually impaired do not participate in the computing profession to their full potential. Awareness of potential career paths and access to adequate preparation remain barriers to students who are visually impaired. Computing courses are becoming commonplace in

many middle and high schools, where topics ranging from networking to programming are taught. However, students who are visually impaired find barriers in such courses in terms of the ability to fully participate in class activities and in sufficient tool support. Even today, many people are not aware that the people who are visually impaired can participate in computing at all.

Project ACE, Accessible Computing Education, focuses on the needs of students who are visually impaired. The phrase, visually impaired, encompasses the spectrum of impaired vision. The term “legally blind” is defined through US law, referring “central visual acuity of 20/200 or less in the better eye with the best possible correction, as measured on a Snellen vision chart, or a visual field of 20 degrees or less” [2]. To better understand the visually impaired population, a statistical portrait will be presented. As of 1995, approximately 1.3 million Americans are considered legally blind [5]. “Of these individuals, 80% (1,040,000) had some “useful vision” (a rate of 40 per 1,000). The other 20% (260,000) had only light perception or less vision (a rate of 1 per 1,000). Half of these individuals were totally blind (130,000), that is, had no light perception (a rate of 0.5 per 10,000).” [6]

The exact numbers of students with visual impairments who pursue computing education is not known, but anecdotal evidence suggests the numbers are very low, as statistics either are only filtered for gender and ethnicity or all disability groups are combined. As of 1999, approximately 1.5 million visually impaired computer users, including those who are blind, use computers regularly [1]. As described earlier, academic preparation in the courses that channel students into computing is low. A major factor in the low participation is the difficulty in accessing information, especially the highly abstract and visually oriented information that is common to computing [8]. Working with computers, in addition to learning about the concepts therein, is a highly visual experience where diagrams and visual elements are commonplace. As such, exploration of computing by young students who are visually impaired is less likely than that of their sighted counterparts. The attitudinal challenges that impact one’s perception of potential career paths affects students with visual impairments much as it does with students from other underrepresented groups.

The goal of Project ACE is to increase participation in computing by the students who are visually impaired. Reaching secondary-school students (grades 7 – 12) is critical as the academic and extracurricular experiences shape educational and career choices after high school. The initial effort in Project ACE is the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGCSE ’08, March 12–15, 2008, Portland, Oregon, USA.
Copyright 2008 ACM 978-1-59593-947-0/08/0003...\$5.00.

ImagineIT workshop, offered to students in grades 7-12 who are visually impaired [4]. The 4-day workshop covered robotics programming, PC hardware, and game development. This scope of this paper is limited to the robotics activities using Lego Mindstorms NXT.

2. SUPPORT FOR STUDENTS WITH VISUAL IMPAIRMENTS

2.1 Differences in Vision

Potential participants in the workshop had a wide variety of vision. Applications for student participants included the type and extent of vision and the use of vision aids for reading and computer use. Students also wrote essays in which many wrote about their interest in the workshop and in technology. Some students had no vision at all, where others could discern some light. Others were closer to the threshold for the definition for being legally blind, where magnification was simply needed.

With the visual profiles of the participants at hand, informed choices were made about the use of assistive technology and the extent to which sound and touch would be used in the activities could be made with confidence. As one of the authors is also visually impaired, the first-hand experience and perspective was also leveraged.

2.2 Instructional Support

While the visual ability of each student varies, there exists a general set of common support mechanisms that will allow for the students to participate in the activities.

Handouts for the students were available in Braille and in large print. While the printing of materials in large print is relatively easy, some reformatting of the document was needed to preserve understandability. Braille printers are expensive and many schools will not have them. Online Braille printing services exist, often with a turn around of about a week. As Braille paper size is different from that of standard paper, the number of lines of Braille that fit on the page will vary from that of the standard or large print version. No diagrams were included in the handouts, but if any diagrams are needed they should be described clearly if a tactile image printout is not feasible.

Assistive technology is not new to students with visual impairments, so asking them about their technology of choice is essential. The students were most familiar with Windows and the popular screen reader and magnification software is only available on Windows. Even with a modest budget, enough copies of the popular screen reader and magnification programs, JAWS and Zoomtext, were available for the students to use at the computer. During the orientation period of the workshop, the software was customized for the students with their assistance. Some students were able to use the built-in contrast and screen resolution adjustment features in Windows with no other technology support needed. A set of speakers was bought for each computer, in addition to a few sets of headphones. Sets of keyboard labels that presented each key in large print and Braille were attached to the sets of keyboards. A simple Braille labeler was also bought to create simple labels for the Lego Mindstorms NXT robot brick ports.

3. THE ROBOTICS ACTIVITY

3.1 Technology Selection

Robots are finding uses in mainstream society and they offer a concrete, tactile mechanism for students to explore their potential. Lego Mindstorms have established themselves as the means to explore robotics and programming in many secondary and university classrooms. The diverse set of sensors (e.g. light, sound, touch) offers students of varying perspectives the opportunity to contribute to the solution of problems. The simplicity and potential of Lego Mindstorms NXT made the technology selection simple. However the programming language and environment was less simple. The Robolab iconic programming language is not accessible to students with visual impairments.

The foundation for the Lego Mindstorms activities was the selection of an accessible language and environment that could be leveraged in a 2-day activity. A strong following of people who have extended the capabilities of Lego Mindstorms has resulted in a diverse set of language support. Due to the likelihood that most participants have little to no programming experience, a C-like language called NXC (Not eXactly C) was selected [7].

While Java is commonly used as an introductory programming language, the NXC language was selected as an appropriate choice to provide a low learning curve and was accessible. The programs the students develop will be relatively small yet provide the desired design challenge. The procedural NXC language has a simple, easy-to-learn syntax that was easily readable by screen reader software. A companion programming environment, BricxCC [4], was selected when found to be compatible with the JAWS and Zoomtext screen reader/magnification software. BricxCC is an open source environment that supports varying robotics technologies, including Lego Mindstorms NXT.

3.2 The Activities

The students know the notions of navigation and orientation, as they can relate to the need for robots to do the same in order to maneuver through a space. A design problem was devised that the students work towards after they learn about the NXC language and the BricxCC environment. A 4-foot by 4-foot maze was created for the robots to navigate.

The activities were designed to follow an iterative development approach over a two-day period. A set of 7 lessons was designed to provide the students with the foundation of the NXC language and the basic capabilities of the robots' sensors. Handouts were prepared along with example source code associated with the lessons. An additional, file with additional subroutines was also developed in order to ease the process of turning the robot. Rather than turning off individual motors, the subroutine for turning only required the number of degrees in the turn and whether the turn was right or left. The tutorial lessons consist of:

1. The robot moves forward for 1 second and stops.
2. The robot moves forward for 4 seconds, turns around, and plays a sound
3. The robot moves forward until touch, turn right
4. The robot repeats the following 3 times: Move forward until it touches something and then turn right 90 degrees.

5. The robot moves forward towards a sound source, touches the sound source and stops.
6. While there is sound, the robot moves forward towards the sound source, touches sound source and then turns left 90 degrees.
7. The robot moves forward toward maze wall, and turns left 90 degrees when the ultrasound sensor detects the wall at 25 cm. After the left turn, the robot plays a sound.

The handouts walk the students through each lesson, including the new concepts and the language features used. The source code is also commented, as shown in Figure 1. The screen reader software was set to read the punctuation in order for the students with low vision to understand the role of punctuation in programming. With each lesson, the students walked through the concepts and how to manipulate the BricxCC environment, in addition to the robot itself. All of the lessons were traversed in 3 to 4 hours by all four teams.

```
#include "Subroutines.nxc"
/**
 * This program moves the robot forward
 until it touches a wall with the touch
 sensor. When it does, it stops, moves
 backwards for one second, then turns to
 the right. It repeats this process three
 times.
 **/

task main()
{
  SetTouchSensor;
  repeat(3)
  {
    forward;
    until(TouchSensor == 1);
    stop;
    backup;
    Wait(1000);
    stop;
    turn(90, RIGHT);
  }
}
```

Figure 1. NXC Code Sample from Lesson 4.

After the tutorial, the design challenge was introduced along with the development process. The challenge for each team was to program their robot to navigate through the maze and locate a sound source. After the maze is exited and the sound source located, the robot needs to indicate this by playing a sound and stop movement. The process of solving the problem was approached in small increments, to allow for controlled progression and chances for feedback. Each team approached the problem in two increments:

1. Navigate Through the Maze using the Touch Sensor
2. Navigate Through the Maze using the Sound and either the Touch or Ultrasound sensor to locate the sound source and play sound

For each increment, the team designed a small part of the program and tested it, usually in the maze itself. Figure 2 shows a team testing their robot in the maze.



Figure 2. Team is testing their robot during Increment 1.

When a defect was found, the problem was discussed in the team and the program was redesigned to fix the defect. For example, the first step in Increment 1 required the robot to successfully move towards the maze wall in front of it and turn right. Then, the program was redesigned to work with the path through the maze with right and left turns. The first increment was usually completed in a brute force fashion before a more generalized solution was designed. As the designed solution became more complex, the student teams became engaged in discussions with multiple approaches. Testing was also taken as a more serious endeavor where hypotheses were made about problems that arose and issues with the hardware were taken into account when designs were revised in the next version. A student is seen working on the team's program in Figure 3.

After the second increment, each team's robot was run through the maze in front of all four teams. The teams cheered for the robots and were supportive of each team's accomplishments. Although not all robots completed the entire design challenge, the process of working together and learning about software development was common to all teams.

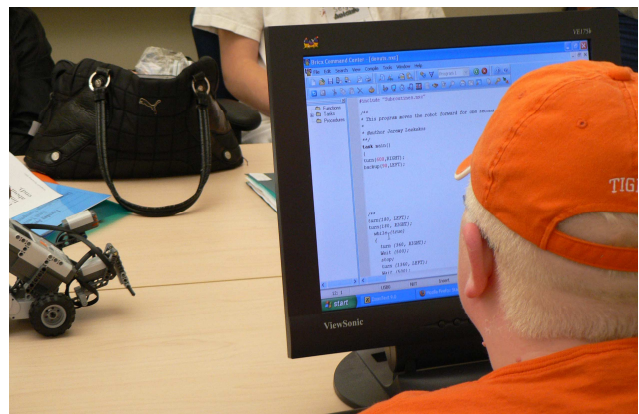


Figure 3. Student working with NXC program in BricxCC.

In addition to the process, students were encouraged to follow roles during these activities in order to encourage participation from all students. The length of the activity and the use of multiple increments, allowed for roles to be rotated. For example the student who entered the program changes was rotated, though all students were interested in testing the robot itself.

4. FEEDBACK

4.1 The Participants

Fourteen students participated in the ImagineIT workshop. The students ranged between 12 and 18 years old. All but one of the participants was male. The ethnic, geographic and socioeconomic backgrounds were diverse. In addition to the initial data gathered in the application, students were given surveys at the end of the workshop. Data was collected to assess the value of the different activities and of the structure of the workshop. All but one student completed the survey. The survey was available in large print and was available on the computer for students who needed the survey read for them. An overview of the results that is relevant to the Lego Mindstorms activity is shared in the next session.

4.2 Initial Results

The students were asked questions about their experience, attitudes and their likelihood that they would pursue computing further.

When asked about the likelihood that the students would take a computer class in their secondary school as a result of the workshop did not change, 7 students responded that they would like to enroll in a computing course. This is not a big surprise as their interest level in computing was rated highly. The number could not have been much higher as some students said that computing courses were not available at their schools.

The students were asked about their interest in robotics and programming using a scale from 1 to 5, with 1 meaning no interest and 5 meaning a high level of interest. The average level of interest was 4.15. Twelve students rated their interest level as a 4 or a 5. The students were then asked about the challenge level of the activities, using a scale from 1 (Difficult) to 3 (Easy). The middle rating, 2, represented that the difficulty level was About Right. Of the 13 students who responded, 8 felt that the activity challenge level was about right while 3 felt it was too difficult and 2 felt it was easy.

As the intent for the workshop and the Lego Mindstorms activity was to make computing appealing and fun, the students were asked about how fun they felt the activity was. The students could choose between three options: Not Fun, Neutral, and Very fun. Of the 13 students who responded, 2 students felt the activity was not fun and 11 students felt that the activity was very fun.

Nearly all of the students' written comments centered on the PC Hardware and Game Development aspects of the workshop.

4.3 Parent Feedback

In addition to student participants, many parents also attended the workshop. Some parents were interested in learning about the programming activities and many were also interested in observing their children work actively in a team with their peers. Sample parent feedback included:

"The workshop was geared specifically to visually impaired teens. This was important as many times in a "regular" classroom setting our kids have to fit in with non-VI kids + are constantly trying to play catch-up. This workshop allowed these teens many opportunities to let them see that a career in computers is very possible."

"..The instructors pretty much let the kids do the work themselves as a group. I think it made the kids feel good about themselves, a sense of accomplishment."

"The workshop gave the kids a lot of hands-on experience with computers that they might not have gotten otherwise (putting a computer together, working with robots, game design) plus they were able to share ideas and learn from other students."

An unplanned side effect of parent participation was the ad-hoc networking. Parents were able to share experiences from a diverse set of school districts and the extent to which they did or did not have access to services that met their needs. The need to further share information and friendship was so desired that a list of parent and student contact information was created and distributed to allow for continued interaction beyond the workshop. While this ad-hoc networking was not a planned part of the workshop, it was an unexpected benefit to the parents of the workshop attendees.

4.4 Lessons Learned

From preparing the activities to conducting the workshop, the authors learned a great deal that can help future renditions of the workshop and others' inclusive outreach activities.

4.4.1 Preparing the Activities

When the activity coordinators and student helpers/volunteers are designing the activities, documentation and version control are essential. The use of CVS to manage all source code, tutorial handouts, activity lesson plans, and related documents was helpful in keeping everyone informed of the latest changes. Administrative access to the computers being used in the activities is also important as the installation of BricxCC, JAWS, and Zoomtext required full access to the computers.

Creating accessible materials and activities is the main focus, but ensuring that they are also appealing to the audience is also essential. The activity and goals need to be something the students can relate to. While sound and touch play a more prominent role in the design challenges, issues with working with these elements must be worked out. Using a sound source in a maze was tricky, especially when people are talking or cheering the robot. So the challenge means that the sound source was turned on when the robot was near the end of the maze. Also, selecting the sound source itself involved some research. Toys that made sound did not work since the sound level was inconsistent and over time the sound was annoying. The sound source used was a tone generator played through a laptop.

While one of the coordinators has had extensive experience working with the visually impaired, the other staff did not have experience. Before the workshop was held, a training session was held to inform workshop staff about strategies needed to work with the visually impaired. A handout was created, and is available upon request from the authors.

4.4.2 Conducting the Workshop

While having a set schedule for the activity was useful, being flexible in order to accommodate students who either needed extra help or who are exceptional was essential. Two teams completed the design challenge with time to spare. So an extra challenge was devised whereby the robot needed to navigate a path using a light sensor instead of the touch sensor. This extra challenge required the students to use their existing skills while learning the nuances of a new sensor.

The success of the activity would also not have been possible without each team having an assistant, whether the assistant was a workshop assistant or one of the authors. This arrangement allows for activity facilitation and ensured that all students were active in their respective teams.

Advice for others who are interested in using Lego Mindstorms NXT robots with visually impaired students also includes:

- Spending time orienting the students to the room, their work area, and the equipment is important.
- Space tables, chairs, etc. with enough space for students with canes to traverse the work area safely.
- The use of the rechargeable battery pack
- The use of Braille labels for the major components of the brick.
- The need for a sighted individual to help students with very low vision with menu navigation on the brick.
- Creating a reference list that will enable the facilitator to direct students who are using the Braille version of the handouts to the correct page quickly.
- The need for a large room or multiple small rooms to control the noise level. In addition to the students, screen readers also are speaking constantly.

5. NEXT STEPS

The workshop and activity set was piloted in Summer 2007. Both the event and the Lego Mindstorms activities were a success. With the positive results, the activities will be expanded to allow for similar activities, including those that can transition students to other computing topics in addition to advanced programming concepts. The use of

Java will also be explored as skills in object-oriented programming can be immediately applied to high school and university course work.

6. ACKNOWLEDGMENTS

The ImagineIT workshop was funded as part of the Accessible Computing Education Project, by the National Science Foundation (Grant #0634319).

7. REFERENCES

- [1] American Foundation for the Blind, "Internet and Computer Usage"; available: <http://www.afb.org/section.asp?SectionID=43&TopicID=224&DocumentID=2313>
- [2] American Foundation for the Blind, "Statistics and Sources for Professionals"; available: <http://www.afb.org/section.asp?SectionID=15&DocumentID=1367Fröhlich>
- [3] BricxCC Command Center Homepage; available: <http://bricxcc.sourceforge.net/>
- [4] ImagineIT Workshop Homepage; available: <http://www.se.rit.edu/~imagine-it>
- [5] National Center for Health Statistics, National Health Interview Survey - Disability Supplement, 1994 and 1995; available: <http://www.cdc.gov/nchs/nhis.htm>
- [6] National Eye Institute, "Statistics on Blindness in the Model Reporting Area, 1969-70," (Publication No. [NIH] 73-427), which is conventionally still used as a yardstick.
- [7] NBC/NXC Next Byte Codes and Not eXactly C; available: <http://bricxcc.sourceforge.net/nbc/>
- [8] Smith, Ann C., Francioni, Joan M., and Matzek, Sam D., "A Java Programming Tool for Students with Visual Disabilities," in *Proceedings of Assets 2000*, Washington D.C., November 2000.