Abstract — Computer literacy classes for non-computing majors are received with less than enthusiastic regard by many students. Viewed as simply another general education requirement toward graduation, these classes do not command the attention we would like from students forming the next generation of professionals and policy makers. This paper profiles an attempt to engage these students by supplementing traditional lectures with a series of demonstrations and experiences specifically designed to expose them to both computing principles and possibilities.

Index — Computers in Society, Computer Literacy, General Education, Technology Education

INTRODUCTION

While computer literacy courses are generally regarded as an integral component of the college general education requirement, there is a large diversity in implementation, scope, and acceptance of these classes. Computer literacy itself is a moving target, changing as computers become more ubiquitous and students subsequently expecting more depth in skill acquisition, as well as more breadth in a topical survey.

Even though computer literacy is a moving target, there are some common components to the typical course. As described by Mason and Morrow [1], computer literacy should encompass two distinct components: "awareness" and "competence". In this context competence implies the ability to apply a variety of "hands on" software skills such as word processing, spreadsheets, and database applications. Awareness, on the other hand, implies knowledge of the impact and implications of computing to both individuals and society.

From an instructor's perspective, it is critical that the approach to teaching computer literacy must engage the students as well as develop skills. In fact, there is some evidence that the approach taken to introduce these subjects can have a direct impact on student attitudes toward computers. For example, Burger and Blignaut [2] report that as students gained experience through one computer literacy course both their computer confidence and computer liking decreased as measured the Computer Attitude Scale instrument. They conjecture this may be a function of skill acquisition rate.

Finally, a computer literacy course must address the immediate needs of the stake-holders: students, instructors, and employers. Bartholomew [3], for example, had employers rank the relative importance of a variety of skills. The results ranked spreadsheet skills first, followed by word processing, database, and presentation skills.

Recognizing that we must address both the skills and awareness components of computer literacy, we have designed a course that devotes approximately half of the instructional time to skills acquisition and half to developing a level of computer awareness. Over time, the skills component has been well defined, including hands on laboratory experience with each of the basic component skills: word processing, spreadsheets, database, and presentation. Since the skills are largely prescribed and consistent with the observations by Mason that the "awareness" aspect of computer literacy needs the most attention, the balance of this report will focus on one attempt to enhance the awareness aspect of a computer literacy class.

A CRAFTED APPROACH

Observing that we must develop a variety of topics from scratch, and that time is limited, we decided to create a class where topics were introduced through a sequence of targeted experiences and demonstrations drawing on internet resources, faculty research, and contemporary culture. To that end, our educational objectives include:

- Basic computer vocabulary
- Insight into algorithms
- A hint of theoretical computing
- Legal, ethical, and societal
- Computing as communication
- Applications of computing in disciplines ranging from medicine to remote sensing.

To reach these instructional goals, we deployed actual demonstrations and/or hands on classroom experience both to engage students and to make them aware of applications beyond those already familiar. In principle, these experiences can expand their horizons when they leave the university, and inform their future work as a result. The following sections describe these experiences in the context of specific instructional objectives.

1 James Wolfer, Department of Computer and Information Sciences, Indiana University South Bend, South Bend, IN, USA, jwolfer@iusb.edu
**BASIC VOCABULARY**

A large part of the class is devoted to building basic vocabulary. Hardware concepts and components ranging from CPU’s to the definition of bits and bytes are discussed. Memory, peripheral components, IO devices, and their interconnection are described and physically demonstrated. Following the mastery of the “language of computing” we move to develop insight as follows.

**Systems**

We introduce Linux, and a discussion of the tradeoff between proprietary and open source solutions for both personal and business computing. We also discuss the role of open source for developing countries in the light of recent computer initiatives, and use this to start a discussion of computer security.

**Theory, Algorithms, and Programming**

While computing theory and programming are not required in the computer literacy course, we would like for our students to have some insight into the foundations of computing. To this end we present the following demonstrations and discussions:

- Lindenmayer systems for plant morphology.
- Demonstration of a simple sort.
- Demonstration of Linear Search
- Demonstration of Binary search.

The sort and the searches are designed to reveal the concept of an algorithm. We do not show programs for doing these operations; instead we show by example how simple steps can solve complicated problems. By observing a search or sort happens, and by describing how a programmer might implement them, students gain insight into the nature of programming.

To instill an appreciation of the foundations of computing, we take a slightly different approach. If, in an informal Turing Machine sense, we describe digital computing as a sequence of interpreted string replacements then we can provide a "feel" for this processing through Computer Graphics and Lindenmayer Systems.

Lindenmayer systems dynamically model plant morphology through a series of production rules applied to given strings, or axioms, over a finite alphabet. In this sense they are reminiscent of formal grammars, but have the added advantage of producing a highly visual result. Following the discussion of Burns [4], we produce on the board a caricature resembling broccoli using a series of production rules. While not explicitly an example of formal computing theory, this provides the flavor of theory to a general audience. It also provides a natural discussion of other string-encoded morphology, including natural encoding in DNA. This then becomes a transition point to computer graphics, as described in the next section.

**Computer Graphics**

While describing computer graphics we reference contemporary motion pictures, particularly those composed of computer generated animation. To gain an appreciation of the nature of graphics primitives, such as the pixel, we provide the following demonstrations:

- Mona Lisa pixel brightness
- Stereographic display monitor
- Genetic-Program Computer Art

To illustrate the pixel, a digital photograph of the famous painting the Mona Lisa is enlarged in real-time until each pixel covers an area large enough to display the numeric value of its corresponding brightness. This pointedly illustrates the relationship between a number and the brightness it represents. This is further illustrated with simple color pictures, describing how the primary colors are composed of three such numbers, and how computer graphics is the art and science of selecting these numbers for display.

We also discuss the implications of having two eyes through stereographic display. Using stereographic monitors, we illustrate true 3D graphics, and use this for discussing how objects are represented in terms of simpler components such as polygons and spheres.
Finally, we use computer graphics to make the transition from a discussion of images to artificial intelligence through a system of genetically programmed art [5]. Images, such as that illustrated in Figure 1, are presented along with the concept of self-programming through simulated evolution. This also ties in with basic biology.

**ARTIFICIAL INTELLIGENCE**

In one sense Artificial Intelligence is the study of problems that are extremely difficult to solve and to which we assume intelligence is required. On the other hand, it also provides a “mirror”, a reflection surface against which to test our own presuppositions. It is in this sense we approach AI through a series of robot demonstrations. Specifically, we provide live demonstrations for the following:

- Khepera II
- BalBot robots
- Remote sensing

The Khepera II robot shown in Figure 2 is a hockey-puck sized robot composed of a computer, two differentially driven wheels, and eight infrared proximity sensors. Driven in an obstacle avoidance mode, these robots traverse a maze. Students interact with the robot by placing barriers in front of them, forcing the robots to alter their behavior in real-time.

The Balbot robots, shown in Figure 3, are dynamically-stable, inverted pendulums which balance through sensing their relationship to the surface on which they stand. Students interact with these robots by introducing minor perturbations to the robot's equilibrium, then observing the robot attempt to recover. The robot is also capable of exploring its surroundings while avoiding walls, people, and other impediments to its progress. Together, the robot demonstrations trigger discussions ranging from the meaning of being human to the nature of machines.
Finally, we introduce a real-life application of artificial intelligence by illustrating the application of artificial neural networks to remote sensing. As shown in Figure 4, Landsat imagery is shown and neural networks are deployed to identify human-imposed structures in the images. This also leads to discussions of the appropriate ethical use of advanced technology whose access is not uniformly distributed among countries. It also provides a transition into the use of these technologies in other domains, such as medicine, described next.

**Medical Applications: Cardiology and Mammography**

To further illustrate contemporary applications of computers, we demonstrate their use in medicine through illustrations of animated echocardiology, and the automated detection of ventricular borders as shown in Figure 5 [6]. The students are exposed to the concepts of computer aided diagnosis, and we tie the discussion into the previous robot experiences by discussing robotic surgery.

We also show images of digital mammograms, describe the types of lesions that occur, and discuss the problems associated with the interpretation, storage, and retrieval of such large image databases. We also use this as a discussion point, exploring issues of privacy, law, and ethics. This also forms the segue for human-computer interaction, described next.

**Human Computer Interactions**

The discussion of computers in medicine, including image analysis and robotics, leads to discussions of the interaction between people and the machines around them. Since the students typically have limited experience with HCI devices - typically mice, keyboards, and joysticks, we have developed a suite of experiences to introduce alternative interactions. Specifically,

- Haptic Mouse
- Phantom Omni
- Speech recognition

The Logitech IFeel mouse, along with custom software [7], is used to explore the texture of the Mona Lisa. Using this interface the students receive tactile feedback coupled to the brightness of the underlying image. We then use this as an additional information channel from the computer to help them to find a lesion in a mammogram. In a very tangible sense they get feel for alternative interfaces.

In a similar vein, we introduce the Sensable Phantom Omni. This six degree-of-freedom, force feedback robot allows a realistic sensation of "touching" objects embedded in visual scenes. Again, this is a novel experience for most students, and substantially expands their experience base. It also begins a discussion of practical uses ranging from virtual clay sculpting to medical training.

Finally, we provide a demonstration of a commercial off-the-shelf speech dictation system. This gives the students insight into both the strengths and limitations of current technology in the context of something they do easily, by virtue of being human, but which is difficult for computers. This also engenders a discussion of the types of problems appropriate for computers.
OBSERVATIONS AND FUTURE WORK

While the demonstrations described seem to create genuine interest among the students, they do not cover the gamut of topics necessary. For example, economic issues, networks and communications, and various specific software products such as accounting systems. While some of them address other computing applications, for example robots as embedded systems, they don't provide a lot of detailed information. These topics are covered in the course, but not via hands-on illustrations. Even with these limitations, however, feedback is generally favorable, and the majority of students find they learn new ways of approaching computing that they had not considered before.

In the future we would like to extend the suite of illustrations to include experiences in data sonification, walking robots, and volumetric medical imaging. We would also like to incorporate a discussion of computers in games since this currently represents a "hot topic" area of computing. Finally, we are considering a medical-specific section of the class to cater to health professionals in the region.

SUMMARY

We believe that we have crafted a series of experiences to augment the traditional lecture and serve to introduce the essence of computing ranging from programming and theory to artificial intelligence. We have described demonstrations including searching, sorting, plant morphology, robotics, haptic human-computer interaction, medical imaging, stereographics, evolutionary computing, and art. Taken together, we believe that these experiences not only build requisite skills, but also provide insight into computing that can inform a student's future work.

References


