Close

CollegeBoard AP Central® for Educators

CLICK TO PRINT PAGE

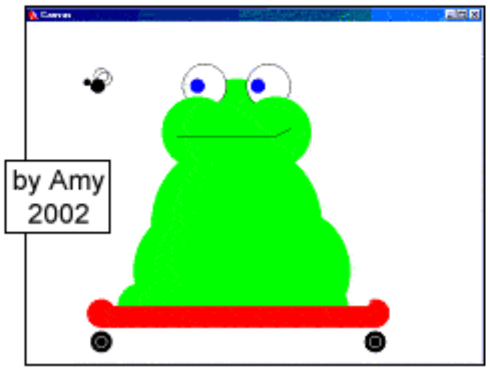
Here is the article in a print-friendly format. Click the button above to print this page.

Supporting Girls in CS by Programming with Graphics

by Karen North
Westside High School
Houston, Texas

Inspiring students to think for themselves will help them on the way to become life-long learners. -- Albert Einstein If you are interested in increasing the enrollment in your computer science class, try an outreach program with art, technology, and middle school teachers. Allow students to independently discover creative thinking through programming graphics.

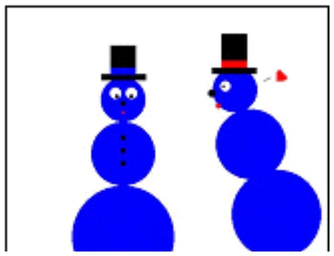
"You gotta have art!" is the headline of an article in the American Teacher March 2003 issue. "It strengthens critical thinking...and has a positive effect on student learning, motivation, interest and attendance." Combine art with the power of programming, which computer science teachers know builds data analysis and problem-solving skills, and students get double the value for their education.



According to the College Board AP* reports, the number of students taking Advanced Placement Exams from 1993 to 2002 has doubled. The percent of females taking math/science AP Exams has grown from 45 percent to 49 percent, yet the number of females taking computer science has decreased from 18 percent to 14 percent in the same time period. In 2002, 66 percent of students taking art AP Exams were female. To increase the enrollment of females in computer science, one of my solutions is to introduce girls to programming through the arts.

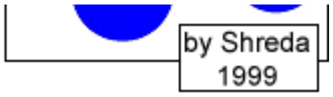
Jennifer and Chelsea Westside High School 2003

If girls are never exposed to programming, how can they ever know if they like it? In order to attract underrepresented groups to study computer science, I have presented workshops on Graphic Programming at the American Association of University Women Expanding Your Horizons conferences and Texas Computer Education Association conferences. To support those who attend these conferences and other interested teachers and students, I have developed an online Graphic Programming module modeled after my Technology Systems Applied Educational Systems modules. Both the lessons located at http://girltech.cs.rice.edu/Participants/knorth/TS/programming.htm and the DrScheme programming software are free. This is a three-week unit that can be incorporated into an art or math class as an independent unit or into a technology class.



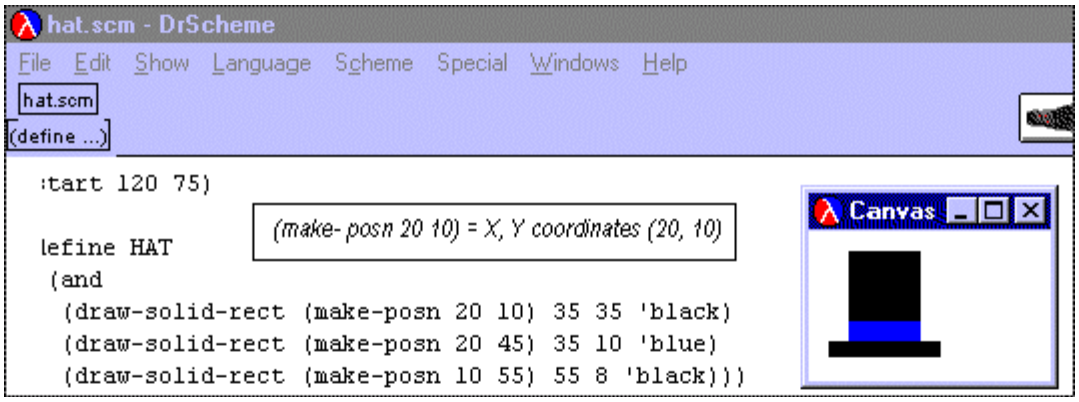
I have found that girls are attracted to the aesthetics of a programming language that is easy to code and debug and to content that is based on art. I observed this to be true when I used LOGO to teach geometry on an Apple IIE in the late 1980s. Now I am developing a series of modules using DrScheme and graphics to help build the domain knowledge needed to be successful in a computer science class. I am also working on a Pre-AP Vertical-Teaming curriculum with middle school technology teachers.

In my programming modules, I have beginner programming students draw pictures on



the computer by plotting points. The purpose of creating graphics by plotting points is to practice using built-in Scheme functions to understand how drawing commands work. By studying the patterns of the points, these examples can then be used in future lessons to create variable functions. This process is also used in the teaching of algebra, which starts in the primary grades with arithmetic. When students first learn to add, they use blocks and fingers as examples to understand the process. Students then build on this primitive operation by adding multiple times. Following this same learning method, when students first learn computer science, they use coordinate points, which are lists of two points, in the same manner as blocks and fingers.

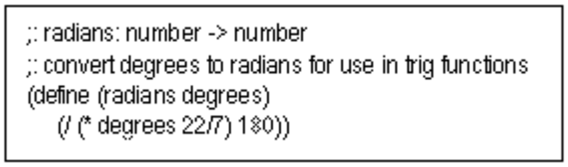
In the graphic programming module, students use the built-in functions draw-solid-disk, draw-circle, draw-solid-line, draw-solid-rect, and make-posn. These functions require color, size, and position parameters. The locations of the coordinate points are lists of two points, the X and Y values. In DrScheme, these lists are called positions (posn). Students create examples that can then be turned into functions in more advanced programming. Creating examples is step 2 in the design recipe used to plan programs found in the free online book, How to Design Programs. I find students often asking, "Is there an easier way?" when repeating similar code. This discovery method promotes the use of functions when they observe what is changing and what is constant. Once the use of variables becomes natural, recursive functions can be used to create loops to repeat a process.


hat.scm - DrScheme
File Edit Show Language Scheme Special Windows Help
hat.scm
(define ...)
(make- posn 20 10) = X, Y coordinates (20, 10)
Canvas

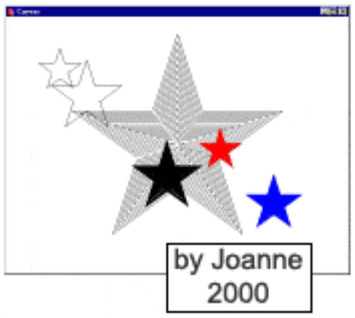
```
:tart 120 75)

lefine HAT
 (and
  (draw-solid-rect (make-posn 20 10) 35 35 'black)
  (draw-solid-rect (make-posn 20 45) 35 10 'blue)
  (draw-solid-rect (make-posn 10 55) 55 8 'black)))
```

The syntax of the code in DrScheme is so simple that students do not get overly frustrated with debugging mistakes. The error messages motivate them to discover solutions. When the girls are successful with simple programs, they are empowered to tackle other problems, such as creating a five-pointed star of any size or position using trigonometric functions. More examples are available at www.knorth.info.



```
;; radians: number -> number
;; convert degrees to radians for use in trig functions
(define (radians degrees)
   (/ (* degrees 22/7) 180))
```



Starting programming with examples of pictures using points plotted on graph paper helps girls understand patterns. This planning step is part of the design recipe found in How to Design Programs. This practice makes the next step of using functions and variables much easier. This helps the normally underrepresented female population of Information Technology students build the brain connections needed for higher algebraic computation. By approaching problem solving through graphics, I find girls motivated to work through the difficulties in algorithm development that turn students away from the study of computer science.

If you are interested in learning more about this outreach program to include computer science programming in middle school technology, math, or art classes, please contact me at knorth@cs.rice.edu or knorth@houstonisd.org.

References:

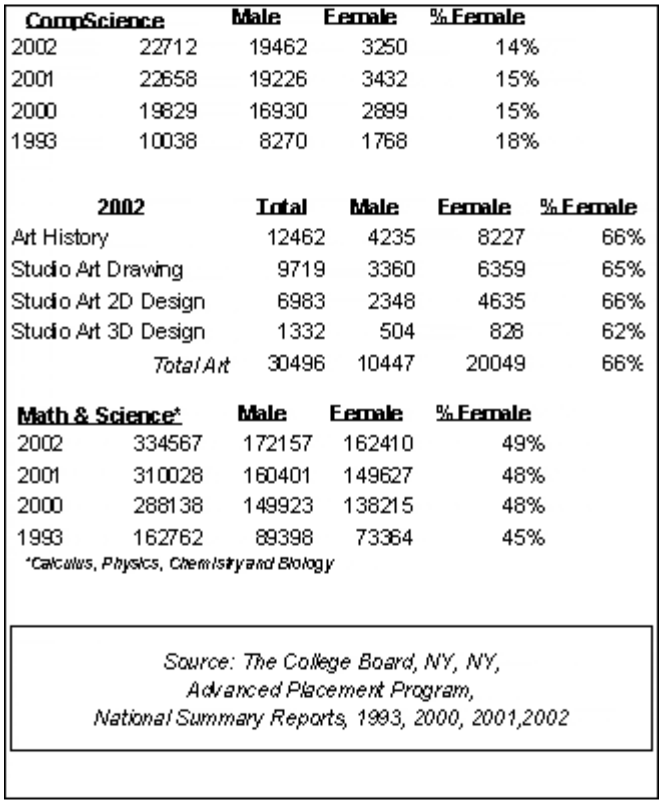
How to Design Programs
http://www.htdp.org/

The TeachScheme! Project
http://www.teach-scheme.org/

AES Technology Systems Modules
http://www.aeseducation.com/

AAUW Expanding Your Horizons
http://www.aauw-whc.org/

*AP College Board Statistics
http://apcentral.collegeboard.com/repository/ap02_national_summary_19132.xls



CompScience		Male	Female	% Female
2002	22712	19462	3250	14%
2001	22658	19226	3432	15%
2000	19829	16930	2899	15%
1993	10038	8270	1768	18%

2002	Total	Male	Female	% Female
Art History	12462	4235	8227	66%
Studio Art Drawing	9719	3360	6359	65%
Studio Art 2D Design	6983	2348	4635	66%
Studio Art 3D Design	1332	504	828	62%
Total Art	30496	10447	20049	66%

Math & Science*		Male	Female	% Female
2002	334567	172157	162410	49%
2001	310028	160401	149627	48%
2000	288138	149923	138215	48%
1993	162762	89398	73364	45%

Calculus, Physics, Chemistry and Biology

Source: The College Board, NY, NY,
Advanced Placement Program,
National Summary Reports, 1993, 2000, 2001,2002