

# National Center for Women & Information Technology

## PROMISING PRACTICES

### Peer-Led Team Learning (Case Study 2)

#### Retaining Women through Collaborative Learning



K-12 Education



Undergraduate



Graduate



Lower course drop rates and higher grades are the common outcomes achieved through peer-led team learning (PLTL) in computer science. In addition to advantages for PLTL participants, peer leaders also benefit from this experience, e.g., gaining greater confidence to continue in CS. For example, at the University of Texas at El Paso, weekly training sessions for peer leaders focus on deepening their understanding of cooperative team and

professional skills such as conflict management, shared leadership, and techniques for dealing with adversity. These weekly sessions improve the effectiveness of the PLTL sessions, the retention of women, and the development of leaders. One training activity has peer leaders role play a workshop session. The other peer leaders and faculty mentors, who have expertise in cooperative learning techniques, critique the scenario. These training activities allow students to dig deeper into understanding their role in building effective learning communities.

Some institutions offer weekly two-hour workshops for students in introductory CS courses. Recruitment for these workshops targets women, minority students, and students from small rural high schools. Each workshop has five to eight students led by a well-trained undergraduate. The students work together on interesting group exercises, helping and learning from each other.

Susan Horwitz, at the University of Wisconsin-Madison, provided the sample group exercise for PLTL in computer science. This exercise is designed to improve understanding of several concepts in Java programming:

- What happens when objects are declared and created, and when methods are called?
- What is the difference between copying from one variable to another when the variable is an object and when it is a primitive type?
- What is the difference between changing values of variables that are objects and changing values of variables that are primitive types?

#### Steps in the PLTL “Car Class” Activity

1. Group members read through the Car class definition provided by their leader.
2. Each student chooses one variable that she will represent.
3. Students act out the role of their variable as specified in **the code fragment to the right**:

#### RESOURCES

For more sample exercises and information about the NSF-funded PLTL consortium in computer science, visit: <http://pltlcs.org>.

**Case Study Contributors:** Susan B. Horwitz and Ann Q. Gates

**NCWIT offers practices for increasing and benefiting from gender diversity in IT at the K-12, undergraduate, graduate, and career levels.**

*This case study describes a research-inspired practice that may need further evaluation. Try it, and let us know your results.*

**ncwit.org** Authors | Lecia Barker and J. McGrath Cohoon  
Copyright © 2007-2008

NCWIT Investment Partners: National Science Foundation, Avaya, Microsoft, Pfizer, and Bank of America

#### PLTL “Car Class” Code

```
Car myCar, yourCar, anotherCar;
int oldSpeed, currSpeed;

myCar = new Car("beep");
yourCar = new Car("honk");
anotherCar = myCar;

currSpeed = myCar.getCurrSpeed();
yourCar.changeSpeed(7);
anotherCar.changeSpeed(20);
currSpeed = myCar.getCurrSpeed();

myCar.blowHorn(2);
yourCar.blowHorn(3);
anotherCar.blowHorn(4);

oldSpeed = currSpeed;
myCar = yourCar;
currSpeed = myCar.getCurrSpeed();

myCar.changeSound("ooga");
myCar.blowHorn(currSpeed/5);
yourCar.blowHorn( yourCar.getCurrSpeed()/2 );
anotherCar.blowHorn( myCar.getCurrSpeed()/10 );
anotherCar = myCar;

class Car {
    /******
     * data members
     *****/
    private int currSpeed;
    private String hornSound;

    /******
     * public methods
     *****/
    /* constructor */
    public Car(String sound) {
        currSpeed = 0;
        hornSound = sound;
    }
    /* changeSound: change the horn sound */
    public void changeSound(String newSound)
    {
        hornSound = newSound;
    }
    /* blowHorn: blow the horn;
     * parameter numTimes tells you how many
     times
     */
    public void blowHorn(int numTimes) {
        while (numTimes > 0) {
            System.out.
println(hornSound);
            numTimes--;
        }
    }
    /* changeSpeed: change speed */
    public void changeSpeed(int milesPerHour)
    {
        currSpeed = currSpeed + milesPerHour;
    }
    /* getCurrSpeed: return the current speed */
    public int getCurrSpeed() {
        return currSpeed;
    }
}
```

# National Center for Women & Information Technology

## PROMISING PRACTICES

### How Do You Retain Women through Collaborative Learning?

with Case Study 2



K-12 Education



Undergraduate



Graduate

**Collaborative learning environments require** that students work together on formal or informal learning activities. For example, collaborative learning occurs when students work in pairs on programming assignments; when small groups of students discuss possible answers to a professor's question during lecture; and when students work together outside of class to learn new concepts. Collaborative learning is distinct from projects where students "divide and conquer" the work for an assignment because it requires that they actually work through problems together, that they engage in intellectual talk with each other.

**A long tradition of research** shows that well-managed collaborative learning environments improve outcomes for all types of students. Outcomes include:

- improved critical thinking
- increased retention, especially for women advancing from the introductory to second course
- appreciation of diversity
- development of social and professional skills

In computer science, collaborative learning environments can improve retention and learning for men and women, and seems to shrink the gender gap in these outcomes. For example, pair programming is one form of collaborative learning with evidence of increased retention of both female and male undergraduates. Other forms of collaborative learning that could succeed in computer science include Peer-Led Team Learning and conversational classrooms.

Collaborative learning environments must be carefully planned and managed by instructors. For example, research shows that when there is too large a gap in collaborators' experience or knowledge, the benefits of collaborative learning disappear. Tasks must also be "shareable" for true collaboration to occur. Most capstone courses in computing education include project learning; however, student collaboration should be introduced early, often, and in both graded and un-graded situations to give undergraduates greater experience and to avoid early socialization that computing is a career in which people work alone. Using collaborative learning in an educational system that rewards individual work can be a challenge for faculty who need to assess individual outcomes while leveraging the benefits of collaboration. Most university campuses have resource centers that will work with faculty to integrate collaborative learning in ways that ensure individual accountability. These examples demonstrate that implementing collaborative environments can be complex. Nevertheless, the benefits to students, faculty, and industry appear to be worth the cost.

#### RESOURCES

For information on Peer-Led Team Learning, see the CCNY website at: <http://www.sci.ccny.cuny.edu/~chemwks/index.html>. It offers a guidebook, some sample problems, references on papers on PLTL, and more.

#### PEER-LED TEAM LEARNING IS A FORM OF COLLABORATIVE LEARNING

Peer-Led Team Learning (PLTL) is a collaborative and active learning technique that forms students in a course into a community of scholars and leads them to take responsibility for their learning. It involves teams of six to eight students that meet weekly in a workshop with a trained peer leader who is under direction of the instructor. During the meeting, the group engages in interesting problem-solving exercises. According to a description of peer-led team learning provided by the PLTL Project at the City College of New York, project evaluations identify six key workshop features:

1. All students in the course must attend the workshops, which are a regular component of the course. This feature encourages students to view the workshop as important to their learning, and integrates workshop activities with course lectures.
2. Course instructors are closely involved with both workshops and workshop leaders. Instructors prepare and review workshop materials and preview problems with peer leaders.
3. Workshop leaders successfully completed the course, are familiar with the assigned workshop problems, are well-trained in teaching and learning techniques and leadership of small groups, and are closely supervised.
4. Workshop materials are challenging at an appropriate level, integrated with other course components, and encourage active and collaborative problem solving.
5. Physical space, allocated time, and other organizational arrangements promote learning.
6. The institution supports innovative teaching.

Instructors must invest time and energy to initially organize PLTL instruction and to assemble workshop materials. After this initial investment, the amount of work involved with a PLTL course is often less than a standard course because fewer students require help during office hours.

NCWIT offers practices for increasing and benefiting from gender diversity in IT at the K-12, undergraduate, graduate, and career levels.

Visit [www.ncwit.org/practices](http://www.ncwit.org/practices) to find out more.

**ncwit.org** Authors | Lecia Barker and J. McGrath Cohoon  
Copyright © 2007-2008

NCWIT Investment Partners: National Science Foundation, Avaya, Microsoft, Pfizer, and Bank of America