Among students who completed
the class, those who paired produced
significantly better programs than
those who worked alone.

were assigned a partner from their list. Those who stated no preference were randomly assigned a partner. Whenever possible students remained with the same partner throughout the quarter, however, due to schedule changes and drops, a small number of partner reassignments were necessary. As a result of hardships such as heavy work schedules or living far from campus, 17 students across the three pairing sections were permitted to program alone for various reasons. Data from these students was combined with the data from the students in the non-pairing section.

Although each student was assigned to one 90-minute lab time per week, most programming assignments were completed outside of scheduled lab time. The labs functioned primarily as teaching assistant office hours. There were no specific in-lab assignments and attendance was not mandatory. Programming assignments were scored for functionality and readability. Along with each assignment, students submitted a log indicating the amount of time they spent on the assignment (pairing students were asked to differentiate between time spent driving, reviewing, and alone), their level of confidence in their solution, how much they enjoyed working on the assignment, and how satisfied they were with the process.

Regardless of whether they completed assignments in pairs, all students took exams independently. The final exam assessed students' knowledge of programming concepts and their ability to write new code. We collected information about students' SAT scores, the courses they took over the following year, and their

major declarations a year after taking the class.

An important assumption of this study was that all four course sections were similar in terms of students' academic preparation to succeed. We found no difference in the SAT math scores among the four sections. We did find that the average SAT verbal score for one of the three pairing sections was lower than the score for the other two pairing sections. However, the difference was not significant when compared to the non-pairing section nor was there a significant difference between the pairing sections as a group and the non-pairing section. Because the difference was only between pairing sections it seemed acceptable.

One of the key hypotheses tested by our study was the following:

Women who program in pairs will have higher retention rates than women who program independently.

Specifically we wanted to know if using pairing as a learning tool for beginning programmers would influence course completion rates and subsequent computer programming course-taking behavior, both in terms of attempts and pass rates, and students' decisions to major in computer science-related fields.

A comparison of students who used pair programming with those who didn't indicated that pairers were significantly more likely to remain in the course through the final exam (90.8%) than were non-pairers (80.4%). Among just those who took the final exam, the difference in pass rates between pairing