

Pair Programming in CS112

When working on programming assignments, whether in lab or elsewhere, you will use the “Pair Programming” approach. In pair programming, **two programmers share one computer**. One student is the “driver,” who controls the keyboard and mouse. The other is the “navigator,” who observes, asks questions, suggests solutions, and thinks about slightly longer-term strategies. The two programmers switch roles about every 15 minutes. Research has shown that working in pairs can make you much better at programming than working alone. The resulting work of pair programming nearly always outshines that of the solitary programmer, with pairs producing better code in less time.

“[Pair programming] makes learning programming faster and more fun. I have had previous experience working both alone and with partners. I definitely agree that working with partners is more profitable.” – CS student

To “learn the do’s and don’ts” of pair programming and to see pairs in action, view this entertaining YouTube video, *An Introduction to Pair Programming*, created by Laurie Williams at North Carolina State University:

https://www.youtube.com/watch?v=rG_U12uqRhE.

CHOOSING PROGRAMMING PARTNERS IN CS112

In CS112, you will start working on the weekly programming assignments in lab. These assignments will typically have *programming exercises* that you (mostly) complete in lab, and extended *programming problems* that you complete outside of class. You will always work with a partner in lab, and for Assignments 1 and 2, we ask that you complete the entire assignment with your lab partner, who will be different for the two assignments. Starting with Assignment 3, you should complete the programming exercises with your partner in lab, but can then work on the programming problems with a different partner, who can be from a different lab section. (Time will be set aside in lecture to help arrange partners.)

You and your partner will need to arrange times to meet outside of class. We expect everyone to be flexible and professional in arranging those times as necessary. If your schedule is highly constrained, you may want to explore possible meeting times with your prospective partner before you commit to the partnership.

Studies of pair programming practice suggest that pairs are most productive when the partners start at about the same level. You may, however, be paired with another student whose skills are different. This happens more often than not, as no two people have an identical skill set. The differences may be great or small, but this is exactly like most real-world working situations. Part of accomplishing a task is to get the most out of each member and make each member stronger and more productive on subsequent tasks.

Students bring different strengths to the process, regardless of how much experience they have had with programming. Both experienced and inexperienced students will need to draw on their reasoning and problem solving skills. A more experienced partner may sometimes feel frustrated or slowed down by a less experienced partner, but the experienced partner still benefits from the teamwork in many ways. The less experienced partner’s requests for clarification often uncover flaws in an approach or solution; the exercise of providing a clear explanation solidifies and deepens the explainer’s own understanding and the teamwork and communication skills they gain have great value in both the academic realm and the job market.

“My partner had never coded anything before so I was able to teach her a little bit about how it worked. The teaching bit helped me a lot with understanding the labs and passing the exams.” – CS student

PAIR PROGRAMMING-IN-A-BOX

The Power of Collaborative Learning

The less experienced partner may feel that questions hold the other partner back or that there is no benefit to participating actively, but pair programming studies show that paired work is consistently better than work the stronger partner does alone. It is each partner's job to understand the whole task; that means asking questions when necessary and answering them when possible.

It may be instructive to read some students' partner evaluation comments from courses that used pair programming; they give a picture of what good partnerships are like (and a few disasters, too). These comments are from an introductory programming course taught at the University of California, Irvine Bren School of Information and Computer Sciences: <http://www.ics.uci.edu/~kay/courses/i41/hw/evalcomments.html>.

DEALING WITH DIFFERENCES

If you believe your partner is not participating appropriately in pair programming (e.g., they do not attend lab, do not keep in touch, or do not come to meetings prepared to work) please first address your concerns to your partner, and try to agree on what should be done to make the pair programming experience work well for both of you. If that approach is not successful, explain the issues to an instructor, who will work with you and your partner to improve the situation.

If your partner drops the course or otherwise stops participating, notify one of the instructors immediately so they can pair you with another student. You may be asked to join another partnership or, as a last resort, to work alone until a suitable partner can be found.

HOW PAIR PROGRAMMING AFFECTS YOUR GRADE

Your attendance and participation in lectures and labs is required in CS112. For your assignment work, you are expected to work collaboratively with a partner using the pair programming method described in this handout and practiced in lab. You and your partner will receive a single grade on each assignment. Being fully engaged in your team programming work provides valuable preparation for the in-class exams that you complete on your own!

SEEK CLARIFICATION

Pair programming is shown to help, not hinder, your success in this course. It is important that you understand the processes and expectations up front so you can gain the most benefit. If you are unsure of any of the aspects of pair programming and how it is implemented in this course, please talk to an instructor.