Pair Programming in CS111

PAIR PROGRAMMING BASICS

When working on programming assignments, in the lab or the classroom, 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 20 minutes. Working in pairs should make you much better at programming than would 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 video about pair programming from North Carolina State University: *An Introduction to Pair Programming for Students* [33].

You will work with a different partner for each team-based lab. We expect students to choose their partners. If you prefer not to choose your own partner, or cannot find a partner, let your teaching assistant know; she or he will find a partner for you. You and your partner must report the partnership to your teaching assistant no later than the end of the lab during which the assignment is given.

You will likely 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, explore possible meeting times with your prospective partner before you commit to the partnership.

CHOOSING YOUR FIRST PARTNER

If it is absolutely infeasible for you to work with a partner, see the instructor. If the instructor determines that there is no other option, she or he may give you permission to work alone. This is rarely allowed. If applicable, you must report your solo status to course staff by the same date that pairs report their partnership.

You should try to pick a partner whose experience and skill level with programming is similar to your own. This may not always be possible and it is sometimes hard to compare skill levels, but students tell us (and other data support) that pairs are most productive when the partners start at about the same level.

Still, people often pair up with another 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 HIM A LITTLE BIT ABOUT HOW IT WORKED. THE TEACHING BIT HELPED ME A LOT WITH UNDERSTANDING THE LABS AND PASSING THE EXAMS." – CS STUDENT

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 a selection of students' partner evaluation comments from similar courses that have used pair programming; they give a picture of what good partnerships are like (and a few disasters, too). Read *Pair Programming Evaluations* from the University of California, Irvine Bren School of Information and Computer Sciences [42].

DEALING WITH DIFFERENCES

If you believe your partner is not participating appropriately in pair programming, 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 your teaching assistant or the instructor, who will work with you and your partner to improve the situation.

If your partner drops the course or otherwise stops participating, notify your instructor or teaching assistant immediately so he or she can pair you with another student. In rare cases 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

Only one submission is required from each team. However, each student is also required to submit a reflection on their team work individually. Students will be docked a few points if they are determined to be poor partners. In special circumstances, when a team member hinders the success of the completion of the assignment, separate submissions from team members will be allowed. If you are in this situation, please discuss your case with the instructor.

SEEK CLARIFICATION

Pair programming is shown to help, not hinder, your successful completion of the introductory 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 the course, see your instructor right away.