Questions to ask students
in CS 3S and CS 9ABCDEFG

Quizzes

Which of these problems were hardest for you?

Did you have any trouble understanding any of the questions?

Explain your solution to this question.

Why did we ask this quiz question?

Suppose the problem were changed slightly in the following way [indicate]. What would be the answer for the new problem?

Case study quizzes and assignments

Did you read the case study? Which study questions did you answer?

In a paragraph or so, summarize the case study.

What aspects of the case study were hard to understand?

How did you go about studying the program?

What did we do in writing the program that you'll do in the future?

What did we do in writing the program that seemed wrong or unnecessary?

Understanding programming assignments

What does the assignment require?

What kind of output should the following input produce?

What kind of input would produce the following output?

Why do this assignment?

Reflecting on design

Describe (in English) how your program solves the problem.

Draw a diagram of the organization of your program.

What other ways to organize the program did you consider, and why did you reject them?

How did you approach the problem?

What difficulties or blind alleys did you encounter in understanding the problem?

What difficulties or blind alleys did you encounter in designing the program?

What did you learn from writing this program about things that you'll continue to do in the future?

What did you learn from writing this program about things that you'll be sure to avoid in the future?

What's the best thing about your program?

What's the worst thing about your program?

Program analysis

Explain in English how this part of the program works.

In which sequence do your subprograms get called?

What values must these variables hold here?

How much can you say about the arguments that [a particular subprogram] is given while the program is executing? Are there any "legal" values for the arguments that will never occur? Why or why not?

Given a particular value for a variable, what should result?

What would your program do when given the following data?

What would make your program crash?

Suppose the problem was changed slightly in the following way [indicate]. How would your program change?

Reflecting on development

What bugs did you encounter?

How did you convince yourself that your program was bug-free?

In what order will you (or did you) implement and test the procedures of the program? Why?

What test cases will you (or did you) use? Why are they convincing?

Debugging

What evidence do you have that the program isn't working?

How does the program behave incorrectly?

What's the simplest test data that exposes the error?

Where is the last place you're sure your program was correct? How do you know?

Which parts of your program are you positive are correct? Why?

What routine(s) do you think could contain the bug, and why?

What additional information do you need to find the bug? How should you generate this information?

Debugging guidelines, by John Dalbey

1. Create a comfortable environment.
2. Attempt to identify the bug covertly. If you can, determine why the student is missing it. Try to anticipate what the student needs to know to find the bug.
3. If the student asks questions about an uncertainty, insist that (s)he formulate the question as a testable hypothesis.
4. If the student seems to have the concept correct but is overlooking a detail, give a reminder to "be suspicious" and continue.
5. If the student seems uncertain about the operation of a language feature, provide a reference to the text.
6. If the student finds the bug at any step, see if (s)he knows how to fix it, and help if necessary. Ask the student to summarize what (s)he got out of the session.