

Survey on Programming Attitudes

EECS 168 Programming 1

This survey is designed to evaluate your programming experience and your perceptions and use of testing and design practices.

KUID

What is your KU ID? _____

Approach Used

What approach did you use on the projects completed in class so far?

	Test-First	Test-Last	Neither
Project 4 (three-dimensional points)	_____	_____	_____
Project 5 (area calculation; polygons)	_____	_____	_____

Time Spent

How many hours do you think you spent completing each project?

Project 4 (three-dimensional points)	_____
Project 5 (area calculation; polygons)	_____

Answer the following questions for **Project 4** (three-dimensional points):

Confidence of Software Quality

I am confident that the code I wrote for Project 4 is correct.

- strongly agree
- agree
- somewhat agree
- neither agree or disagree
- somewhat disagree
- disagree

Confidence of Software Changes

I am confident that I could make changes to the code I wrote for Project 4 without breaking things.

- strongly agree
- agree
- somewhat agree
- neither agree or disagree
- somewhat disagree
- disagree

Confidence of Software Reuse

I am confident that I could reuse the code I wrote for Project 4 in another future project.

- strongly agree
- agree
- somewhat agree
- neither agree or disagree
- somewhat disagree
- disagree

Automated Testing Use

Did you write automated tests (with assert()) for the programs in Project 4?

- yes, all the time
- yes, but only some of the time
- yes, I tried it once
- no, all my testing was done by hand (run program and look at output or use debugger)

Answer the following questions for *Project 5* (area calculation; polygons):

Confidence of Software Quality

I am confident that the code I wrote for Project 5 is correct.

- strongly agree
- agree
- somewhat agree
- neither agree or disagree
- somewhat disagree
- disagree

Confidence of Software Changes

I am confident that I could make changes to the code I wrote for Project 5 without breaking things.

- strongly agree
- agree
- somewhat agree
- neither agree or disagree
- somewhat disagree
- disagree

Confidence of Software Reuse

I am confident that I could reuse the code I wrote for Project 5 in another future project.

- strongly agree
- agree
- somewhat agree
- neither agree or disagree
- somewhat disagree
- disagree

Automated Testing Use

Did you write automated tests (with assert()) for the programs in Project 5?

- yes, all the time
- yes, but only some of the time
- yes, I tried it once
- no, all my testing was done by hand (run program and look at output or use debugger)

Design Techniques

What do you think is the best approach to design a program?

- Don't design, just write code
- Use visual models like the UML or flowcharts
- Sketch the design in code with class declarations before writing function definitions
- Write out the design in natural language
- Use a combination of visual models and natural language
- Let the design evolve as the code is written; document the design with visual models and/or natural language
- Let the design evolve as the code is written; the code is the design documentation

Attitude Towards Test-First Programming

Test-first programming is the practice by which an automated test case is written before the code is implemented. The implemented code is written to pass the test case. The design of the system emerges as the programmer repeatedly writes tests, then writes the code to make the test pass, then improves both the code and tests in short rapid iterations.

What is your opinion of test-first programming?

- I don't think it would ever work
- I think it might be a good approach on small projects
- I think it might be a good approach on projects where programmers have a lot of programming experience
- I think it might be a good approach on projects where programmers understand the domain well
- I think it might be a good approach on any project

Attitude Towards Test-Last Programming

Test-last programming is the practice by which a test case is written after the code is implemented. The design of the system is usually developed at least partially before any code is written.

What is your opinion of test-last programming?

- I don't think it would ever work
- I think it might be a good approach on small projects
- I think it might be a good approach on projects where programmers have a lot of programming experience
- I think it might be a good approach on projects where programmers understand the domain well
- I think it might be a good approach on any project

Perception of Test-First and Test-Last Programming

Regardless of the approach you used, which approach do you think produces code with fewer defects, test-first or test-last?

- test-first
- test-last

Perception of Test-First and Test-Last Programming

Regardless of the approach you used, which approach do you think produces code that is simpler, more reusable, and more maintainable, test-first or test-last?

test-first

test-last

Perception of Test-First and Test-Last Programming

Regardless of the approach you used, which approach do you think produces a correct solution in less time, test-first or test-last?

test-first

test-last

Perception of Test-First and Test-Last Programming

Regardless of the approach you used, do you think you are more likely to thoroughly test a program with the test-first or the test-last approach?

test-first

test-last

Perception of Test-First and Test-Last Programming

Regardless of the approach you used, for the course project you just completed (Project 5), which do you think was the best approach, test-first or test-last?

test-first

test-last

Why?

Choosing Between Test-First and Test-Last Programming

If you had a choice of writing code with a test-first or test-last approach, which would you choose?

test-first

test-last

Why?

Additional Comments

Is there anything else related to this study that you would like to comment on that we have missed or that you would like to add?