Learning Algorithms: What, How and Why

Why study algorithms and data structures? some quotes here, more in discussion and reflection

“This impact is broad and far-reaching.” -- Princeton CoS226 slides

“For me, great algorithms are the poetry of computation. Just like verse, they can be terse, allusive, dense, and even mysterious. But once unlocked, they cast a brilliant new light on some aspect of computing.” — Francis Sullivan

“I will, in fact, claim that the difference between a bad programmer and a good one is whether he considers his code or his data structures more important. Bad programmers worry about the code. Good programmers worry about data structures and their relationships.” — Linus Torvalds (creator of Linux)

What do you need to learn and improve in cs160? Here are the key points and examples based on BST:

- Important data structures, algorithms and underlying principles
  - Why is BST considered a better, more effective data structure than array or linked list based approaches for general symbol table problems? Are there situations where some simpler data structures, such as sorted array, work better than BST?
  - How to check whether a binary tree is a binary search tree?
  - Why do we expand the basic BST node to add a new instance variable for sub-tree size count? Without this field, which methods would be affected, in what way, and why?

- Algorithmic thinking, analysis and problem solving
  - How to deal with duplicate keys?
  - Given a traversal result of a BST, how to reconstruct the tree? Is such reconstruction always possible? Will there be a unique solution or multiple solutions?
  - How to generate a tree with a specified shape?
  - For BSTs with N keys, how many different tree shapes are there?

- Technical reading, and systematic and disciplined algorithm and code development (useful skills for life-long learning in CS and IT)
  - Critical reading, deep thinking and understanding of technical materials
  - Clear big picture and overall design, thorough analysis and testing, incremental development, attention to details ...
  - pursuit of perfection, efficiency and elegance

- Good work ethics and habits
  - Be prepared for your tasks, do your work systematically and complete it on time

“If I had eight hours to chop down a tree, I would spend 6 hours sharpening an axe.”

“Weeks of programming can save you hours of planning.” — Anonymous

“If you don't think carefully, you might believe that programming is just typing statements in a programming language.” — W. Cunningham

“The first 90% of the code accounts for the first 90% of the development time. The remaining 10% of the code accounts for the other 90% of the development time.” — T. Cargill

- Work hard and be persistent
treat integrity issues seriously, learn and earn your credits the honest and right way
“Fool me once, shame on you; fool me twice, shame on me.” – Anonymous
“Honesty is the first chapter of the book of wisdom.” – Thomas Jefferson

learn from your experiences, including those involving failures

How do students learn these topics and skills?
- Learn them the hard way, through diligent work, deep thinking, insightful reflections ...
- Use reading and problem solving to test, evaluate and improve your understanding

The wisdom in the following quotes still applies when one replaces geometry and mathematics with computer science.

“There is no royal path to Geometry.” Euclid (reply given when the ruler Ptolemy I Soter asked Euclid if there was a shorter road to learning geometry than through Euclid's Elements.)

“Mathematics, you see, is not a spectator sport. To understand mathematics means to be able to do mathematics. And what does it mean [to be] doing mathematics? In the first place, it means to be able to solve mathematical problems.” George Pólya

Here are more quotes from George Pólya about problem solving.

“If you can't solve a problem, then there is an easier problem you can solve: find it."

“Even fairly good students, when they have obtained the solution of the problem and written down neatly the argument, shut their books and look for something else. Doing so, they miss an important and instructive phase of the work. ... A good teacher should understand and impress on his students the view that no problem whatever is completely exhausted.”

“It is better to solve one problem five different ways, than to solve five problems one way.”

“A great discovery solves a great problem, but there is a grain of discovery in the solution of any problem. Your problem may be modest, but if it challenges your curiosity and brings into play your inventive faculties, and if you solve it by your own means, you may experience the tension and enjoy the triumph of discovery.”

Why does CS160 feel so different from other courses? Courses have different natures and many (e.g. cs120 and cs160) are not comparable. Also faculty may have different opinions about course focus and different approaches for teaching. For all full-time CS faculty at Clark, including Prof. Green, Prof. Magee, Prof. Basye and me, I am confident to say that we all care about students deeply, want to see our students succeed, and we teach the courses in the best way we know how.

CS160 has evolved for several years and we use past experiences to further fine tune the course. The overall course structure feels suitable to me for the breadth and depth of student learning in the course. I am sure that cs160 and my teaching can use further improvements, and I welcome your suggestions.

As always, student learning is the ultimate goal in all classes and teaching. I hope that we can work together and make cs160 one of the most important and productive courses for your learning at Clark.