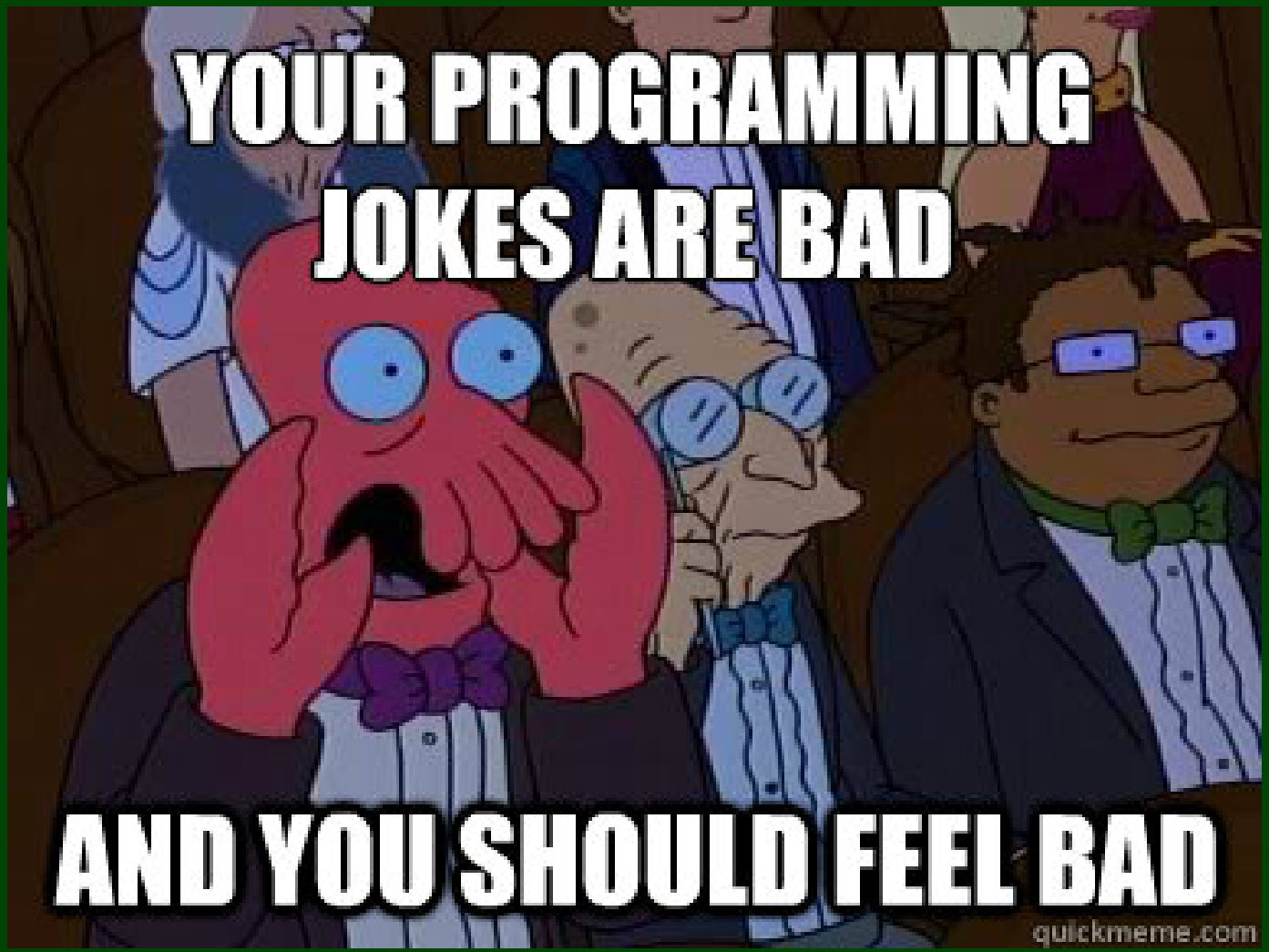
CSE 143X

Accelerated Computer Programming I/II

Goodbye World!



CSE 142 vs. CSE 143: The Big Picture

In **"CSE 142"**, you learned how to use logic, control flow, and decomposition to write programs.

In **"CSE 143"**, you learned to solve **more complex** and **larger** tasks **efficiently**.

CSE 142 vs. CSE 143: The Big Picture

In **"CSE 142"**, you learned how to use logic, control flow, and decomposition to write programs.

In **"CSE 143"**, you learned to solve **more complex** and **larger** tasks **efficiently**.

Big Learning Goals

- Abstraction (implementation vs. client)
- Data Structures (organizing complex data)
- Algorithms (standard ways of completing common tasks)

CSE 142 vs. CSE 143: The Big Picture

In **"CSE 142"**, you learned how to use logic, control flow, and decomposition to write programs.

In **"CSE 143"**, you learned to solve **more complex** and **larger** tasks **efficiently**.

Big Learning Goals

- Abstraction (implementation vs. client)
- Data Structures (organizing complex data)
- Algorithms (standard ways of completing common tasks)

We built some **really cool** programs. And had a lot of fun?

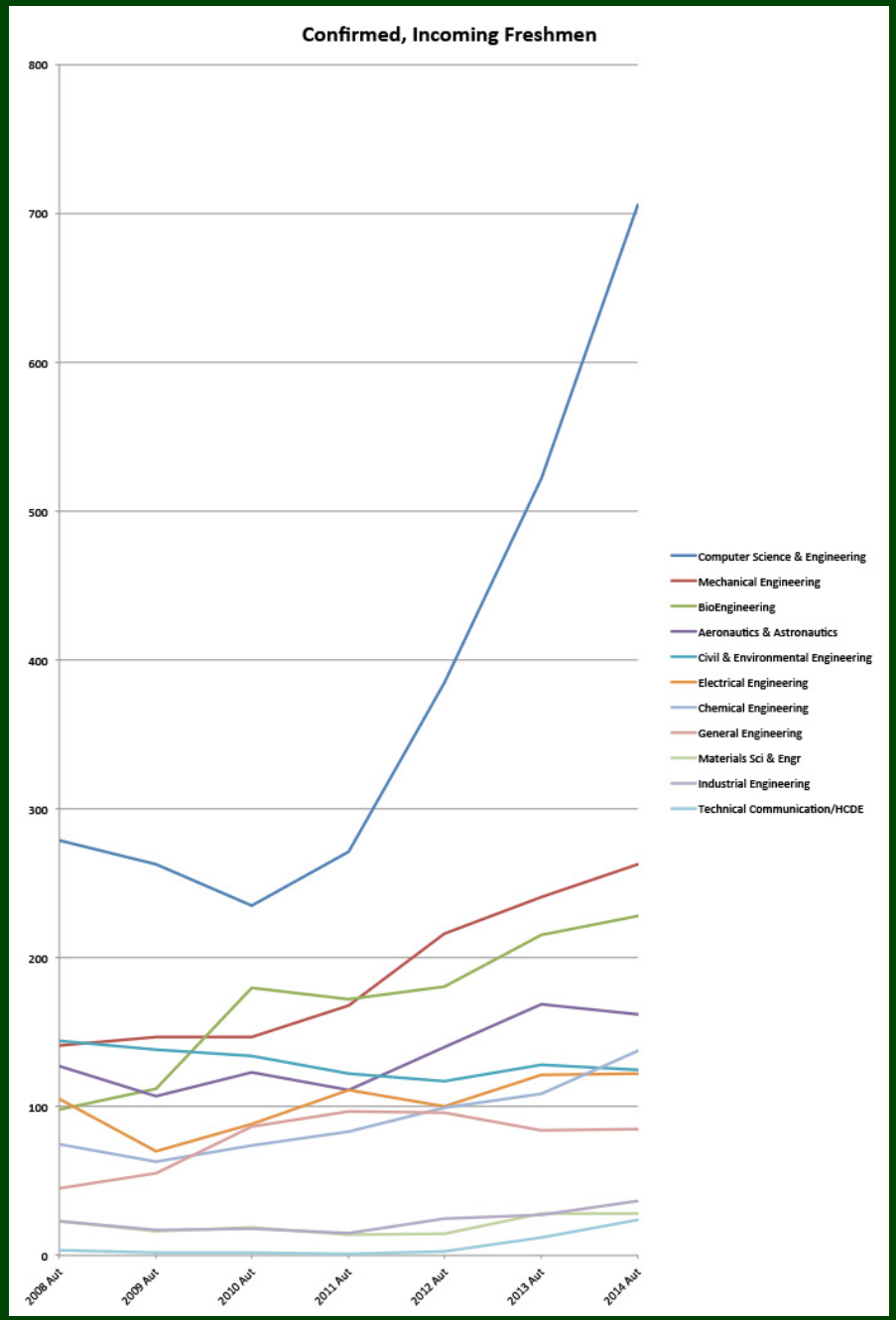
Programmers waste enormous amounts of time thinking about, or worrying about, the speed of noncritical parts of their programs, and these attempts at efficiency actually have a strong negative impact when debugging and maintenance are considered. We should forget about small efficiencies, say about 97% of the time: premature optimization is the root of all evil. Yet we should not pass up our opportunities in that critical 3%.

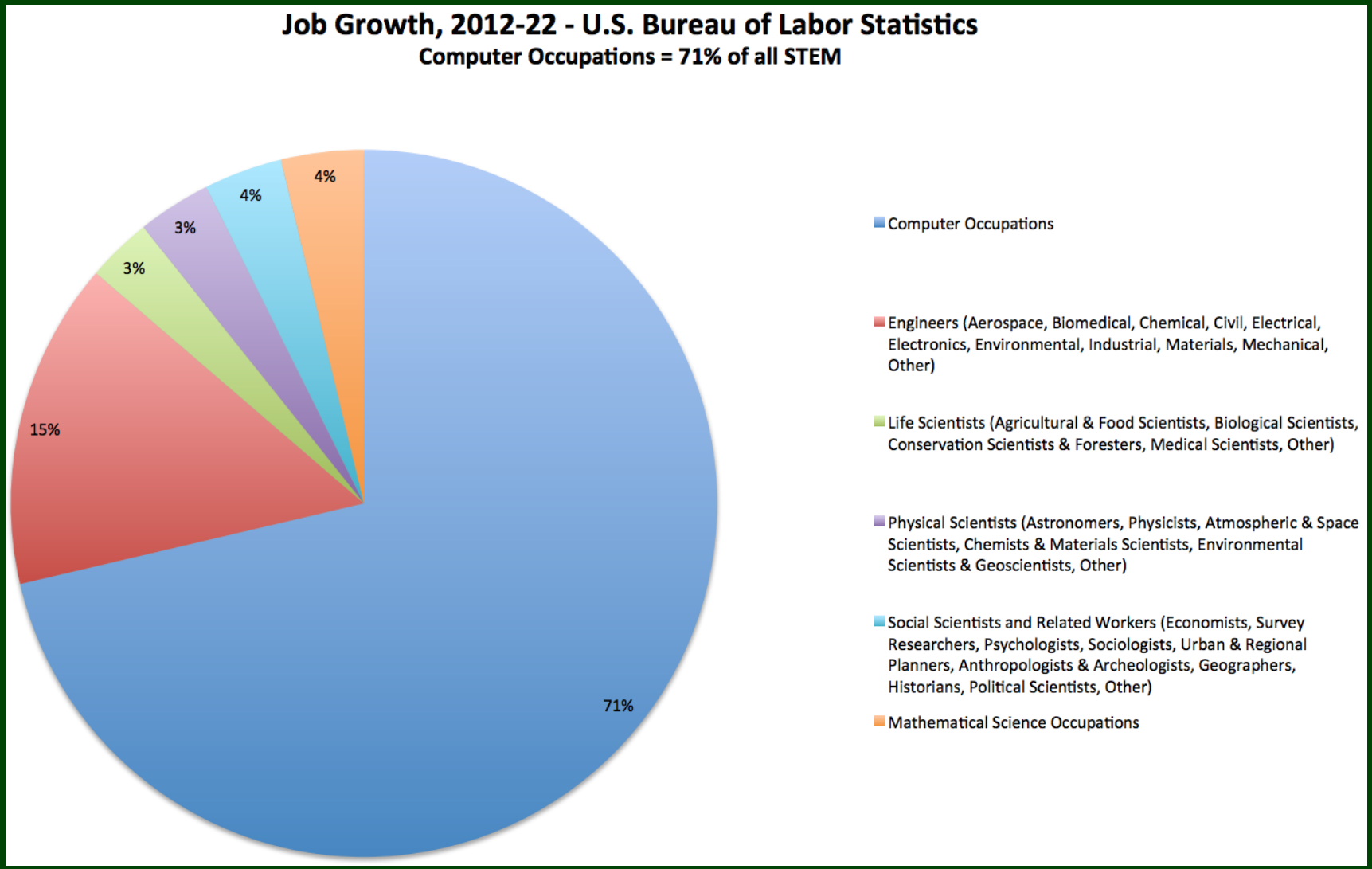
Computer programming is an art, because it applies accumulated knowledge to the world, because it requires skill and ingenuity, and especially because it produces objects of beauty. A programmer who subconsciously views himself as an artist will enjoy what he does and will do it better.

Overview of Topics We Covered Recently

- Lists
- Stacks and Queues
- Recursion
- Sets and Maps
- Grammars
- Searching
- Sorting
- Binary Trees

Computing & Jobs




Job Growth, 2012-22 - U.S. Bureau of Labor Statistics
Computer Occupations = 71% of all STEM
71%
15%
3%
3%
4%
4%
Computer Occupations
Engineers (Aerospace, Biomedical, Chemical, Civil, Electrical, Electronics, Environmental, Industrial, Materials, Mechanical, Other)
Life Scientists (Agricultural & Food Scientists, Biological Scientists, Conservation Scientists & Foresters, Medical Scientists, Other)
Physical Scientists (Astronomers, Physicists, Atmospheric & Space Scientists, Chemists & Materials Scientists, Environmental Scientists & Geoscientists, Other)
Social Scientists and Related Workers (Economists, Survey Researchers, Psychologists, Sociologists, Urban & Regional Planners, Anthropologists & Archeologists, Geographers, Historians, Political Scientists, Other)
Mathematical Science Occupations

Computer Science + Your Interests
=
A Match Made In Heaven

- Foreign Policy: outcome prediction
- Law: evidence summary
- Medicine: smart diagnostics
- Music: hit identification
- Sports: superstar discovery
- Wall Street: high frequency trading

What Should I Do Next?

There are a variety of “next steps” in computer science and programming. Here’s some of them:

What Should I Do Next?

There are a variety of "next steps" in computer science and programming. Here's some of them:

- Learn a new programming language. (Some people find learning new languages fun! The best part is that now that you know Java it's surprisingly easy!)

What Should I Do Next?

There are a variety of "next steps" in computer science and programming. Here's some of them:

- Learn a new programming language. (Some people find learning new languages fun! The best part is that now that you know Java it's surprisingly easy!)
- Learn how to make games/websites/mobile apps/etc. Whatever is interesting to you. . . go for it!

What Should I Do Next?

There are a variety of "next steps" in computer science and programming. Here's some of them:

- Learn a new programming language. (Some people find learning new languages fun! The best part is that now that you know Java it's surprisingly easy!)
- Learn how to make games/websites/mobile apps/etc. Whatever is interesting to you. . . go for it!
- Choose a project to work on that sounds interesting!

What Should I Do Next?

There are a variety of "next steps" in computer science and programming. Here's some of them:

- Learn a new programming language. (Some people find learning new languages fun! The best part is that now that you know Java it's surprisingly easy!)
- Learn how to make games/websites/mobile apps/etc. Whatever is interesting to you. . . go for it!
- Choose a project to work on that sounds interesting!
- Take another course!

What Should I Do Next?

There are a variety of "next steps" in computer science and programming. Here's some of them:

- Learn a new programming language. (Some people find learning new languages fun! The best part is that now that you know Java it's surprisingly easy!)
- Learn how to make games/websites/mobile apps/etc. Whatever is interesting to you. . . go for it!
- Choose a project to work on that sounds interesting!
- Take another course!
- Interview for a job!

What Should I Do Next?

There are a variety of "next steps" in computer science and programming. Here's some of them:

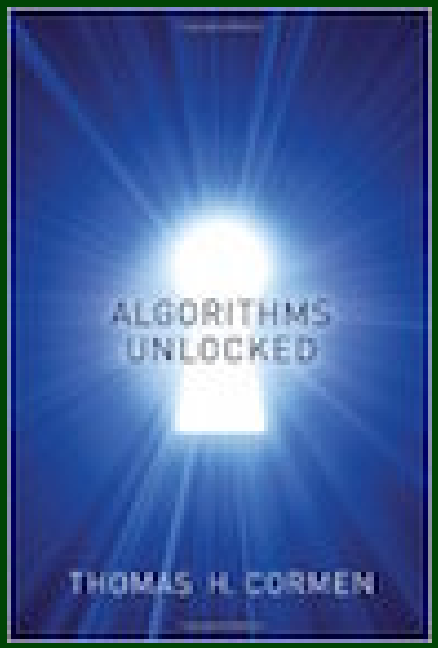
- Learn a new programming language. (Some people find learning new languages fun! The best part is that now that you know Java it's surprisingly easy!)
- Learn how to make games/websites/mobile apps/etc. Whatever is interesting to you. . . go for it!
- Choose a project to work on that sounds interesting!
- Take another course!
- Interview for a job!
- Use programming in your area of interest!

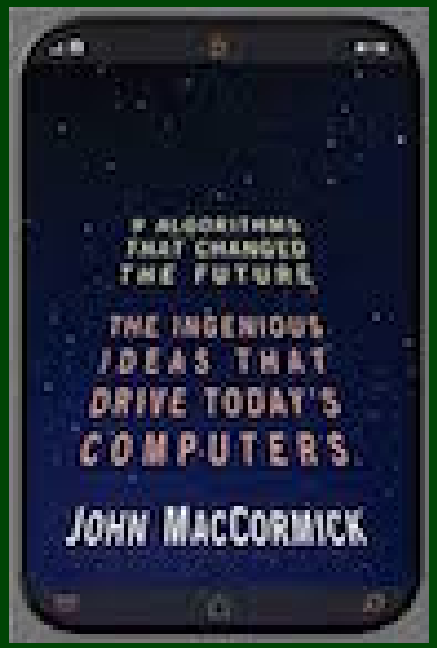
What Should I Do Next?

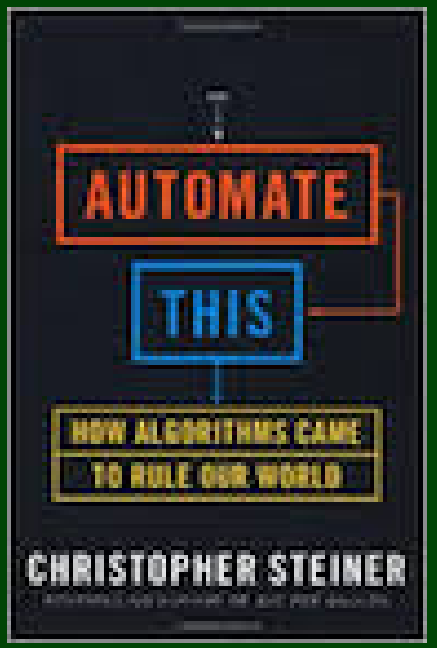
There are a variety of "next steps" in computer science and programming. Here's some of them:

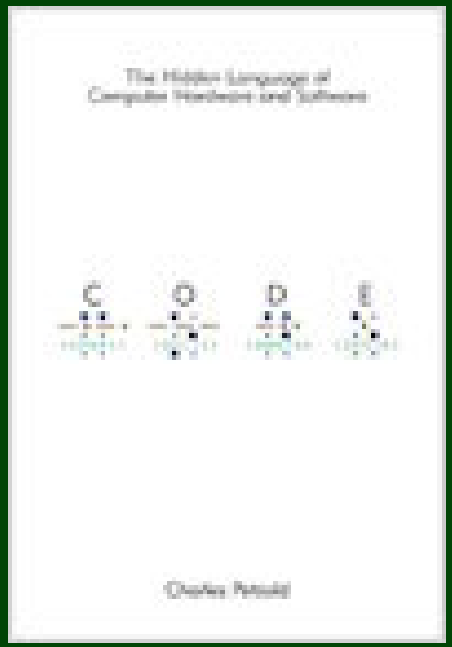
- Learn a new programming language. (Some people find learning new languages fun! The best part is that now that you know Java it's surprisingly easy!)
- Learn how to make games/websites/mobile apps/etc. Whatever is interesting to you. . . go for it!
- Choose a project to work on that sounds interesting!
- Take another course!
- Interview for a job!
- Use programming in your area of interest!
- Learn more theoretical computer science (by reading a book, or something)

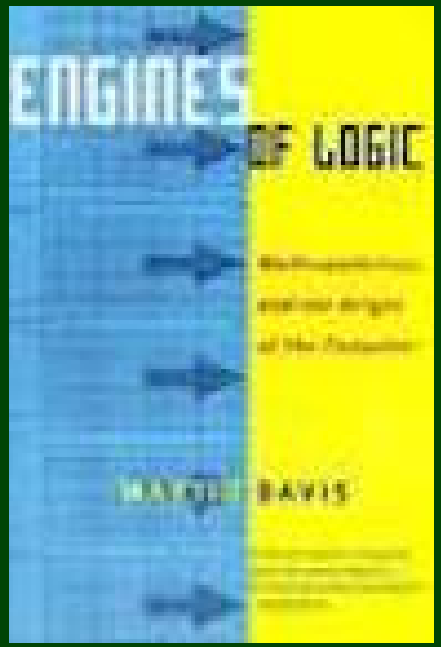
Book Recommendations

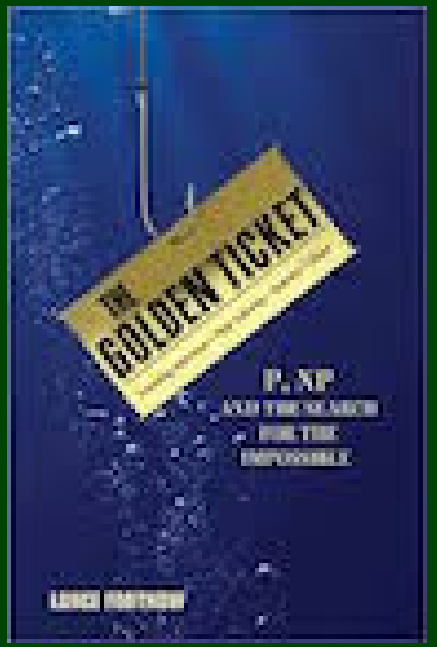












What Project?

When choosing a personal project, the most important thing is that **you think what you're doing is cool**. Here's some suggestions:

What Project?

When choosing a personal project, the most important thing is that **you think what you're doing is cool**. Here's some suggestions:

- Automate something that you do a lot!

What Project?

When choosing a personal project, the most important thing is that **you think what you're doing is cool**. Here's some suggestions:

- Automate something that you do a lot!
- Make one of the HWs from this class cooler!

What Project?

When choosing a personal project, the most important thing is that **you think what you're doing is cool**. Here's some suggestions:

- Automate something that you do a lot!
- Make one of the HWs from this class cooler!
- Write a program that solves your homework for you!

What Project?

When choosing a personal project, the most important thing is that **you think what you're doing is cool**. Here's some suggestions:

- Automate something that you do a lot!
- Make one of the HWs from this class cooler!
- Write a program that solves your homework for you!
- Make a game!

What Project?

When choosing a personal project, the most important thing is that **you think what you're doing is cool**. Here's some suggestions:

- Automate something that you do a lot!
- Make one of the HWs from this class cooler!
- Write a program that solves your homework for you!
- Make a game!
- Write an app and sell it!

What Project?

When choosing a personal project, the most important thing is that **you think what you're doing is cool**. Here's some suggestions:

- Automate something that you do a lot!
- Make one of the HWs from this class cooler!
- Write a program that solves your homework for you!
- Make a game!
- Write an app and sell it!
- Contribute to an open source project!

Java is a perfectly good language, but. . .

Depending on what you want to do, you might want to (or need to) learn another language.

Java is a perfectly good language, but. . .

Depending on what you want to do, you might want to (or need to) learn another language.

- **Python** has very consise and clean syntax. It's useful for whipping up quick programs.
- **Javascript** is the language of the internet.
- **Java** is the language for Android.
- **Objective-C** is the language for anything Apple.
- **C** is the langauge for low-level systems programming.
- **Haskell** is a "functional" programming language. Learn this one if you want a challenge!

CSE Non-Majors

- CSE 154: Web Programming
- CSE 373: Data Structures and Algorithms
- CSE 374: Programming Concepts and Tools (C/C++, Linux, ...)
- CSE 131: Digital Photography
- CSE 460: Animation Capstone (open to all majors)
- INFO, AMATH, DXARTS, ...

CSE Majors

- CSE 311: (Mathematical) Foundations of Computing
- CSE 332: Data Abstractions (Data Structures and Algorithms)
- CSE 331: Software Design and Implementation
- CSE 341: Programming Languages
- CSE 344: Intro to Data Management (and databases)
- CSE 351: Hardware/Software Interface

- Making computers understand language:
  http://nlp.stanford.edu/software/
- Building Games:
  http://lwjgl.org/
- Building Games with Physics:
  http://jbox2d.org/
- Processing Biological Data:
  http://biojava.org/wiki/Main_Page
- Accessing Facebook Data:
  http://restfb.com/

DM4.ME
dm4.me