**Unit 2: Computer Science**

*"Computer science is no more about computers than astronomy is about telescopes." E. W. Dijkstra*

**Computer Science**

[What is Computer Science?](http://computingcareers.acm.org/?page_id=8)

*“Computer science (CS) spans the range from theory through programming to cutting-edge development of computing solutions. Computer science offers a foundation that permits graduates to adapt to new technologies and new ideas. The work of computer scientists falls into three categories: a) designing and building software; b) developing effective ways to solve computing problems, such as storing information in databases, sending data over networks or providing new approaches to security problems; and c) devising new and better ways of using computers and addressing particular challenges in areas such as robotics, computer vision, or digital forensics (although these specializations are not available in all computer science programs). Most computer science programs require some mathematical background.” (*[*http://computingcareers.acm.org*](http://computingcareers.acm.org)*)*

**A. Designing and Building Software**

*Career Path 1: Designing and implementing software. This refers to the work of software development which has grown to include aspects of web development, interface design, security issues, mobile computing, and so on. This is the career path that the majority of computer science graduates follow. While a bachelor’s degree is generally sufficient for entry into this kind of career, many software professionals return to school to obtain a terminal master’s degree. (Rarely is a doctorate involved.) Career opportunities occur in a wide variety of settings including large or small software companies, large or small computer services companies, and large organizations of all kinds (industry, government, banking, healthcare, etc.). Degree programs in software engineering also educate students for this career path.*

Software development includes several phases that we’ll talk more about in our next unit. For now, let’s focus on phase 1 -- the planning phase.

An algorithm is a step-by-step process for solving a problem that begins with a problem statement and its sole purpose is to solve the problem in an accurate and efficient way. Keep in mind that this process occurs before any program code has been written!

To get a visual picture of algorithms, programmers and analysts commonly use one of two tools to demonstrate the algorithm’s program flow: pseudo code and flowcharts.

psuedocode is basically a combination of human-like language (everyday English) and actual programming language. If used appropriately and without heavy dependence on a specific programming language, pseudocode can be a powerful tool for programmers to quickly write down and analyze an algorithm.

Take this problem statement:

***Turn the air conditioning on when the temperature is greater than or equal to 80 degrees or else turn it off.***

Given this problem statement, an algorithm implemented in pseudocode might look like this:

If temp >=80

Turn AC on

else

turn AC off

This pseudocode uses a combination of language and programming syntax to depict the flow of the algorithm; however, if inserted into a C program it would not compile. That’s not the point of pseudocode. It is just a shorthand notation for demonstrating what an algorithm looks like and can easily be transformed into any programming language in stage #2 (Implement the Plan) as described above.

**PRACTICE WITH THE DESIGN PROCESS . . .**

Example 1: Your friend from Europe comes to visit you in America. You are watching the weather forecast. The temperature is given in Fahrenheit, but your friend is used to temperature measures in Celsius. The problem here is that you have to convert the Fahrenheit temperature into a Celsius temperature so that your friend knows what sort of weather to expect. The formula to convert F to C is: C = 5/9 \* (F – 32).

You could write a computer program to perform the temperature conversion. Typically a program has 3 main parts:

1. Take input.

Temp in F

1. Process the input.

Apply conversion formula

1. Provide output.

Temp in C

Example 2: While shopping, you want to add up the prices of the goods that you intend to buy and also calculate the total including the tax.

1. Input:

Prices of the items

Tax rate

1. Process:

Calculate sum of all prices

Calculate tax on the sum

Calculate total (sum of all things plus tax)

1. Output:

Total amount owed

Example 3: Another program could provide you with the name of the month given a number between 1 and 12. For instance, if you gave the program the number 5, it should print “May” on the screen.

1. Input: month number

2. Process: conversion from month number to month string

3. Output: month string

**Discussion Questions**

1. Use pseudocode and write an algorithm that describes the process of brushing your teeth.

Get toothbrush

Rinse toothbrush off

Locate toothpaste

Unscrew toothpaste

Squeeze out toothpaste on toothbrush

Turn water on

Put bristles with toothpaste under water stream

Turn water off

Bring bristles into mouth

Scrub back and forth on teeth

Spit out bubbles

Turn on water

Cup hands and move them under water stream

Collect water

Bring water to mouth

Sip and swish water around inside mouth

Spit water out into sink

Rinse sink down

Dry off mouth

Dry off toothbrush

Put toothbrush away

Close drawer

2. Create a flowchart that describes how to make a PB&J. (Include what happens if you’re out of PB!)

**B. Devising New and Better Ways of Using Computers**

*Career Path 2: Devising new ways to use computers. This refers to innovation in the application of computer technology. A career path in this area can involve advanced graduate work, followed by a position in a research university or industrial research and development laboratory; it can involve entrepreneurial activity such as was evident during the dot-com boom of the 1990s; or it can involve a combination of the two.*

[Early Innovators: Homebrew Computer Club](http://www.cnn.com/2013/10/14/tech/innovation/wozniak-homebrew-kickstarter/index.html)

[Dot Com Boom . . . and Bust](http://www.computerhistory.org/revolution/the-web/20/394)

[Stanford Research (Check out the budget!)](http://facts.stanford.edu/research/)

**C. Developing Effective Ways to Solve Computing Problems**

*Career Path 3: Developing effective ways to solve computing problems. This refers to the application or development of computer science theory and knowledge of algorithms to ensure the best possible solutions for computationally intensive problems. As a practical matter, a career path in the development of new computer science theory typically requires graduate work to the Ph.D. level, followed by a position in a research university or an industrial research and development laboratory.*

[How to Sort Containers Lightest to Heaviest?](http://csunplugged.org/sorting-algorithms)

[Sorting Networks](http://csunplugged.org/sorting-networks)

[Computational Analysis at Carnegie Mellon University](http://www.casos.cs.cmu.edu/)

**D. Planning and Managing Organizational Technology Infrastructure**

*Career Path 4: Planning and managing organizational technology infrastructure. This is the type of work for which the new information technology (IT) programs explicitly aim to educate students.*

[Information Technology](http://computingcareers.acm.org/?page_id=7)

*According to acm.org, career paths 2 and 3 are undeniably in the domain of computer science graduates. Career paths 1 and 4 have spawned the new majors in software engineering and information technology, respectively, and information systems graduates often follow Career path 1, too. Computer scientists continue to fill these positions, but programs in software engineering, information technology, and information systems offer alternative paths to these careers.*

[Information Systems](http://computingcareers.acm.org/?page_id=9)

[Software Engineering](http://computingcareers.acm.org/?page_id=12)

[Computer Engineering](http://computingcareers.acm.org/?page_id=11)

[Mixed Disciplinary Majors](http://computingcareers.acm.org/?page_id=10)