

CS 1050: POWER OF COMPUTING: THINK LIKE A COMPUTER

<i>Semester Hours:</i>	3.0	<i>Contact Hours:</i> 3
<i>Coordinator:</i>	Jong Kwan "Jake" Lee	
<i>Text:</i>	Blown to Bits: Your life, liberty, and happiness after the digital explosion	
<i>Author(s):</i>	ABELSON, H., LEDEEN, K., & LEWIS, H. R	
<i>Year:</i>	2008	

SPECIFIC COURSE INFORMATION

Catalog Description:

Computing as a creative process; Problem solving using abstraction; How data and information create knowledge; Algorithms and computer programs; Internet, digital devices and their impact on society; Privacy and security issues; Computation leading to innovations in other fields.

Course type: **ELECTIVE**

SPECIFIC COURSE GOALS

- I can demonstrate computational thinking practices.
- I can use the core ideas of computer science, such as abstraction, and algorithms.
- I can do simple computer programming.
- I can give an overview of the Internet and its impact to the human society.
- I can discuss the privacy and security issues in using digital devices.

LIST OF TOPICS COVERED

- Connecting Computing (~10%)
 - Introduction to computers, digital devices, the Internet and Big Data
 - The structure of the Internet
 - How computation led innovations in different science fields
 - (Positive and negative) impact of computing and Internet on human society
- Computational Artifacts (~15%)
 - Create a webpage using HTML

- Data representation in digital devices: decimal to binary conversion, binary arithmetic, Base 16, and color (RGB), more
 - What is a computer program?
- Abstraction in Art / Life / Programming (~15%)
 - Use of abstraction in computation or modeling
 - Representing information or knowledge for computational use
- Computational Thinking (~20%)
 - Introduction to Algorithms
 - How to analyze a problem to design the algorithm for solving the problem
 - Example algorithm: How to repeat an operation by using the “conditional loop” concept
 - How to write a program implementing an algorithm (e.g. use of the loop construct)
- Understanding popular algorithms (~15%)
 - How to find stuff: search algorithms (linear and binary search)
 - How to arrange stuff: sorting algorithms (insertion sort and selection sort)
- Special hands-on programming experience (~13%)
 - Introduction to hands-on programming environments (e.g., robots, MIT Scratch lab), which enable programmers to create interactive stories, games, etc.
- Secrecy and Privacy in the digital world (~12%)
 - Data theft and unintended information disclosure
 - Cryptography for data encryption.
 - Privacy issues related to the use of the Internet, online social networks, mobile devices, and the likes