



## Overview

It is finally time for your first lab. Woo hoo! Well, not so much a lab per se, but something to do this week. We are getting rather close to our first official lab, so it's a good idea to understand the data that your program will use.

So, for this first assignment, you are going to convert some values to their binary forms.

## Part 1: Integers

First, let's have you convert some decimal numbers to their binary format. Show your work for credit.

1. This first number will fit nicely into a single byte.

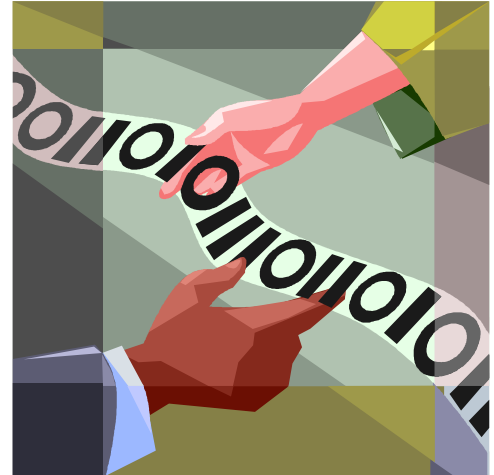
42

*The "Ultimate Answer" from Hitchhikers Guide to the Galaxy*

2. The second number is a tad larger, and will require 2-bytes (16 bits to represent)

451

*A classic novel that appears to be coming true.*



## Strings

Text is stored as a series (an array) of bytes. In the case of classic ASCII, each character is stored with one byte. For this section, you are going to hand-convert two strings to their hexadecimal (and binary) representations.

Here is an example:

H	48	0100 1000
e	65	0110 0101
r	72	0111 0010
k	6B	0110 1011
y	79	0111 1001

For your homework assignment, you are required to put each character on a different line. Each character will be by the hexadecimal and binary equivalents. Your turn-in must be hand-written. Show your work (if needed). You don't have to draw the lines shown above.

3. Convert the following text to a series of bytes.

**Sacramento State**

*Go Hornets! Stingers Up!*

4. Convert the following text to a series of bytes. Use your name rather than YourName. For example, if your name is SpongeBob, convert the following text: **My Name is SpongeBob**. If you have a long name, you can stop after 8 characters.

**My name is YourName**

*Use your actual name rather than the literal text "YourName".*

## **How to Turn In**

Homework will be due one week after being assigned. You will submit your assignment in-class. All work must be **done by hand**.

Do **not** upload them to Canvas. Do **not** e-mail. Assignments uploaded to canvas will **not** be graded.