CDA 4101 **Project 1: Hamming Codes** **Spring 2017**
Pestaina **Due: Tuesday, February 7**

Project 1 replaces Test 1 scheduled for Jan 31. It counts for 25 points of your course grade.

Overview
Implement the Hamming Coding and Decoding Algorithms for Single-Error Correction.

Part A (65%)
A start-up file *Hamming.java* is provided. You must implement two methods:

```
public static int[] encoding(int[] dataBits, boolean evenParity)
public static int[] decoding(int[] codeBits, boolean evenParity)
```

The `int[]` parameters dataBits and codeBits store the binary digits of the data to be encoded, and the encoded bits respectively. Parameter `evenParity` is **true** to indicate even parity, or **false** to indicate odd parity. Only single-error correction is required.

Hint: Use an *ArrayList* for the internal implementation of the code-word.
Hint: Implement a helper method to generalize the calculation of checksums for any check-digit 1, 2, 4, … and any parity whether odd or even.

A completed test program *HammingTester.java* is provided; some sample output is given on the class webpage.

Part B (35%)
Write a client program to

- Input a text message
- Encode the characters of message using the `encoding()` method
- Introduce random single-bit errors into the encoding
- Decode the encoded message using the `decoding()`, correcting all errors

The text message should be composed of the printing 7-bit ASCII characters only. The class *ASCII_Chars* provides static methods to convert between these **char** values and binary digits. It is provided in the file *ASCII_Chars.java*.

A start-up client program *HammingClient.java* is provided.