ECE 3020 Homework 3
Due Date: Friday, February 1, 9:05 AM

1) Write a recursive sorting function that works in the following way. When given an array of n integers to sort, the function ignores the first element of the array, recursively sorts the remaining n-1 elements, and then goes through the array to place the first element in its proper place shifting the other elements to the left as appropriate. In your output, show the contents of the array immediately after the return of each recursive call. A sample output for the input [1 -5 47 32 -17 -21] is:

```
[1 -5 47 32 -17 -21]
[1 -5 47 32 -21 -17]
[1 -5 47 -21 -17 32]
[1 -5 -21 -17 32 47]
[1 -21 -17 -5 32 47]
[-21 -17 -5 1 32 47]
```

Turn in your source code along with three sample outputs, one of which should be the above example.

2) Hamming codes are the most efficient single-bit-error correcting codes. A Hamming code with c check bits can have up to $2^c - c - 1$ data bits. Bits in the code word are numbered from 1 to $2^c - 1$ and the check bits occupy the power of 2 positions, i.e. positions 1, 2, 4, 8, ... The remaining positions are filled by the data bits. The i^{th} check bit is a parity over all bits where the bit position in binary has a '1' in the i^{th} bit. So, the first check bit is a parity over positions 1, 3, 5, 7, …, the second check bit is a parity over positions 2, 3, 6, 7, …, etc. For $c = 3$, a Hamming code word has the form $c_1\, c_2\, d_1\, c_3\, d_2\, d_3\, d_4$. For example, the Hamming code word for data 0110 is 1100110. Compute the 4-bit data words corresponding to the following 7-bit words, some of which are valid Hamming code words and some of which have single-bit errors inserted. Show your work.

1100110

0100011

1101001

1011101

1110001