**Name:**

**Advanced Programming in C++**

**Parity**

**Lab Exercise 3/19/2020 Stardate: 73213.11**

In this lab you will explore the mathematical concepts of parity and invariants. You will use your programming skills to solve these problems. You are to print your documented source code including a sample output.

The term parity refers to the evenness or oddness of the number of bits set to ‘1’ in a packet. If the number is even, we say that the packet has even parity; otherwise it is odd.

If you store binary data or are sending information over a communication line, some errors might occur. A simple way of detecting error is through the use of parity. Usually data is broken apart into relatively short sequences of bits called *codewords* or *packets*.

For example, if we were to use 8-bit packets, we might add a 9th bit called the parity bit. For example if a packet to be transmitted was 11010111 and we wished to use odd parity, the 9th bit would be set to 1 and the packet would become 110101111. If we were to be using even parity, the 9th bit would be set to 0 and the packet would become 110101110.

1. Write a program that allows the user to enter a string and the string is converted to an even parity stream of binary bits. Consider using the countBits function that you previously wrote.
2. The Universal Product Code (UPC) has 12 digits. The checksum is calculated as follows: we take the sum of all of the digits of all the odd positions, starting from the left (first, third, fifth, etc.), multiply it by 3, and add the sum of all of the digits in the even positions (second, fourth, sixth, etc.). If it is a valid UPC, the checksum (total) must be evenly divisible by 10. For example, 072043000187 is a valid UPC. Because

0\*3 + 7 + 2\*3 + 0 + 4\*3 + 3 + 0\*3 + 0 + 0\*3 + 1 + 8\*3 + 7 = 60

which is evenly divisible by 10. Write and test a function isValidUPC(s) that takes 12 digits and returns true if it represents a valid UPC.

1. As you know, the Fibonacci sequence can generate very large integers. They are also very computational intensive to calculate. In order to calculate the nth Fibonacci number we can utilize Benet’s Formula which says:



What is the largest Fibonacci number that can be calculated using this method? Hint: consider using the long long int data type. Answer: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_