Programming Assignment 3

All functions should be in a main.cpp file. Only include: **`iostream, string, cmath,`** and **`stdexcept.`**

Task 1

Create a function called pad() that takes a constant string reference parameter and an integer parameter. The string parameter represents a signed binary number. If the integer parameter is greater than the length of the string parameter, return a padded version of the binary number whose length equals the integer parameter. The padded value should be that of the sign bit. You can assume the string parameter is a binary string consisting of only 0s and 1s.

Example: **`pad("1010", 8) -> "11111010", pad("0111", 6) -> "000111"`**

Task 2

Create a function called trim() that takes a constant string reference parameter. The string parameter represents a signed binary number. Return a string that represents the binary number with any padded sign bits remove. Only one sign bit should remain. You can assume the string parameter is a binary string consisting of only 0s and 1s.

Example: **`trim("11111010") -> "1010", trim("00000111") -> "0111"`**

Task 3

Create a function called binaryToDecimal() that takes a constant string reference parameter. The string parameter represents a signed binary number. Return an integer that represents the binary number as a decimal number. If the number is negative, DO NOT USE two's complement. Perform the weighted sum technique. However, instead of adding the most significant bit's weight to the accumulated sum, subtract it. This will work on positive and negative binary numbers. Ensure your string parameter consists of only 0s and 1s.

Example: $111010_2 = -(1 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$

$= (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) - (1 \times 2^5)$

$= 16 + 8 + 0 + 2 + 0 - 32$

$= -6$

Example: $001010_2 = -(0 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$

$= (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) - (0 \times 2^5)$

$= 0 + 8 + 0 + 2 + 0 - 0$

$= 10$

Task 4

Create a function called hexToBinary() that takes a const string reference parameter. The string parameter represents an unsigned hexadecimal number. Return a string that represents the binary conversion of the string parameter. Ensure your string parameter consists of only characters 0-9, A-F, or a-f. Hint: Use a look-up table.

Example: **`hexToBinary("7A7") -> "011110100111"`**

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111

Task 5

Create a function called subtract() that takes two constant string reference parameters. The string parameters represent signed binary numbers. Return a string that represents the binary number difference of subtracting the second parameter from the first. Remember, subtraction is just addition after changing the sign of the subtrahend. 10 – 5 = 10 + (-5)

Example: **`subtract("01010", "01100") -> "11110"`**