

## Homework 2

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

All runtime tables and formulas should be in one file submitted to BrightSpace. If done on paper, upload a photo or scan to BrightSpace. All code should be uploaded to your GitHub class repository.

### Task 1:

- Define the function **toBase()** that was displayed in class. The function should convert an integer to a binary string.  
Example: `toBase(73) => "1001001"`
- In a separate file, or on paper, show the runtime table and runtime formula for the `toBase()` function.
- Specify the Big-O notation.

### Task 2:

- Create a function called **isPrime()** that takes an integer parameter and returns true if the number is prime, false if otherwise. A prime number is a number greater than 1, that has only two factors: 1 and itself.  
Example: `isPrime(2) => 1`; `isPrime(7) => 1`; `isPrime(25) => 0`;
- In a separate file, or on paper, show the runtime table and runtime formula for the `isPrime()` function.
- Specify the Big-O notation.

### Task 3:

- Create a function called **letterCount()** that takes a constant string reference parameter. The function should return an array (pointer to the array) that contains the individual letter count for each letter in the string. The array should be of size 26. Each index from 0 to 25 represents the letters 'A' – 'Z' respectively. Your solution should be case insensitive: 'A' and 'a' are considered the same. If there are 5 'C's and 7 'c's in the string parameter, index 2 of the array should be assigned 12.

Example:

```
letterCount("hello world!")=>{0,0,0,1,1,0,0,1,0,0,0,3,0,0,2,0,0,1,0,0,0,0,1,0,0,0}
```

- In a separate file, or on paper, show the runtime table and runtime formula for the `letterCount()` function.
- Specify the Big-O notation.

### Task 4:

- Create a function called **sumArray()** that takes an integer array parameter. The function should add all the values in the array and return the sum. The array parameter will only ever be an array created by the function in Task 3. That is to say, the length of the array parameter will always be 26.  
Example: `sumArray(letterCount("hello world!")) => 10`
- In a separate file, or on paper, show the runtime table and runtime formula for the `sumArray()` function.
- Specify the Big-O notation.

### Task 5:

- Create a function called **union()** that takes two integer vector parameters. Each vector represents a set of unique characters. The function should return a vector containing all integers from both parameters.  
Example: `union({1,2,3,4},{2,4,6,8,10}) => {1,2,3,4,6,8,10}`
- In a separate file, or on paper, show the runtime table and runtime formula for the `union()` function.
- Specify the Big-O notation.