# C++ Programming
## Programming Assignment 2

Instructions:

1. **Read instructions carefully!**
2. Use C++ syntax only, C syntax won't be allowed.
3. Always use braces to define blocks.
4. Organize your code well with proper indentation and a single statement per line.
5. Always insert a space before and after each operator, e.g. `a = b + c;` not `a=b+c;`
6. Use meaningful variable names following conventions using *camelCasing*.
7. Comment your code to clarify your thoughts if needed but do NOT comment every single line of code!
8. You may not use any library functions unless explicitly specified.
9. Name each program problem1.cpp, problem2.cpp, etc. and upload each file to Canvas.
10. Validate all input and re-prompt as long as an invalid value is entered! You don't need to validate the type.

**Plagiarism**: Plagiarism of any kind will not be tolerated! Plagiarized assignments will be reported to the Campus Disciplinarian.

### Problem 1 Compute Sum 5 points

Write a program that prompts the user to enter a positive integer. Use a loop to compute the sum of all integers between 1 and the number entered.

**Example**:

```
Enter a positive number: -8
Invalid!
Enter a positive number: 8
Sum: 36
```

### Problem 2 Sea Level Rise 6 points

Write a program that computes the rise of ocean levels for the next x years where x is an integer larger than 2 entered by the user. Use #include <iomanip> to properly align the output match the formatting in the example below.

**Example:**

```
Enter a value (>2): 1
Invalid!
Enter a value (>2): 8
Years | Sea level rise
----------------------
    1 |  1.5mm
    2 |  3.0mm
    3 |  4.5mm
    4 |  6.0mm
    5 |  7.5mm
    6 |  9.0mm
    7 | 10.5mm
```

**Problem 3** **Checking Account** **19 points**

Write a program that simulates a checking account. Initially, the program prompts the user to open an account and deposit an initial balance of at least $20. Afterwards, the program repeatedly shows a menu with the following options:

1. Depositing money
2. Withdrawing money
3. Viewing the current balance
0. Exit

Split your program using functions. Define the following prototypes and implement them below the main:

```
// shows menu, prompts for and returns valid option
int promptOption();
// prompts for amount (>0) to deposit and adds it to the balance
void deposit(float &balance);
// prompts for amount (>0) to withdraw and
// subtracts it from the balance
 void withdraw(float &balance);
```

**Example:**

```
Open Account
How much would you like to deposit? (at least $20): 10
Invalid!
Enter a value (at least $20): 25
----------------
Menu
1 Deposit money
2 Withdraw money
3 View balance
0 exit
Select an option: 4
Invalid!
Select an option: 1
--DEPOSIT--
Amount: 0
Invalid!
Enter an amount larger than 0: 5
----------------
Menu
1 Deposit money
2 Withdraw money
3 View balance
0 exit
Select an option: 2
--WITHDRAWAL--
Amount: -10
Invalid!
Enter an amount larger than 0: 10
```

```
----------------
Menu
1 Deposit money
2 Withdraw money
3 View balance
0 exit
Select an option: 3
Balance: $20.00
----------------
Menu
1 Deposit money
2 Withdraw money
3 View balance
0 exit
Select an option: 0
Goodbye!
```