**COMP2710 – Frequently Asked Questions**

**Homework 2**

1. How do I tell the program about the decimal number format?

   **Answer:**
   ```
   cout.setf(ios::fixed);
   cout.setf(ios::showpoint);
   cout.precision(2);
   ```

2. Do you see any other mistakes in my source code?

   **Answer:** You need to design your own test cases to check potential bugs in your program.

3. Do numbers greater than 999 need to have commas separating their digits? For example, would our program have to output $2,789.98 or would $2789.98 be okay?

   **Answer:** You do not need to have commas separating digits if a number is greater than 999. In the above case, $2789.98 is acceptable.

4. I understand that monthlyInterest equals to balance*monthlyRate. If monthlyRate is 1.5%, I think that instead of balance*1.5, it is actually supposed to multiply 1000*.015 to get the correct answer.

   **Answer:** You are right. Suppose your monthly interest rate is 1.5%, then monthly interest can be computed as below:

   monthlyInterest = balance * 0.015;

5. In Homework 2, how can I get my double values to print with the appropriate number of zeros in the decimal? For example, when trying to print the initial balance (1000) I need to print $1000.00 but I am just getting $1000.

   **Answer:** You may use the "magic formula", which was introduced in class. Please refer to the lecture slides posted on Canvas. See also Question 1.

6. Is there a specific way we should round our decimals for the dollar amounts displayed in the table/used in calculations to get the correct dollar amounts? Not sure whether we should round, ceiling, floor, etc. or if that matters a great deal or not.

   **Answer:** You only need to setup the format using the following code:
   ```
   cout.setf(ios::fixed);
   cout.setf(ios::showpoint);
   cout.precision(2);
   ```

7. Do I need to figure out the alignment of the columns or is the result I get quite ok?

   **Answer:** No. You do not have to. Nevertheless, you are encouraged to use “setw()” (see the sample code in question 8 below) to address the alignment issue.

8. Do we need to make the spaces uniform like they are in the example? Different lengths for month, money amounts, etc make the spacing skewed a bit for the output. if the output is correct, is that something that should be cleaned up? If so, do you have any suggestions as to fix the spacing irregularities?

   **Answer:** You may use the `setw(num_chars)` function is used to set the width of the text to be output on the console window. Please feel free to try my sample code to learn this function. (see also alignment.cpp)

```
#include <iostream>
#include <iomanip> //std:setw
using namespace std;

int main( )
{
    float float_a = 10.53;
    float float_b = 6329.07;

    cout << "No setw() ..." << endl;
    cout << float_a << "   " << float_a << endl;
    cout << float_b << "   " << float_b << endl;

    cout << "Add left and setw() ..." << endl;
    cout << left << setw(10) << float_a << float_a << endl;
    cout << left << setw(10) << float_b << float_b << endl;

    return 0;
}
```

   Do not forget to add “left” in the cout line.

9. if the interest is higher than the monthly payment, do you want us to use cout to create an error message such as "Error: monthly payment is less than interest." Or do you want the program to just exit immediately?

   **Answer:** It is a good idea to issue an error message like what you proposed.

10. For my rate, since I applied the decimal format to two decimal places for all of my other values, my rate is showing up as 1.50. Could you please give me some advice as to how to force this variable to only calculate to one decimal place?

    **Answer:** If you can't solve this problem, the TA will not deduct any point due to this format issue. In case you want to address this issue, you simply repeatedly switch between `cout.precision(2) and cout.precision(1)`