

A. While loops

Write a program that will take the average of a series of entered floating point numbers. Specifically, your program should:

1. Ask the user to enter the first number. (It should be of type double.)
2. Then continues to ask for more values to be input, with the option of stopping the inputs by entering a value of 0.
3. The program calculates and outputs the average of all of the entered values. (The output should be nicely formatted.)

B. For loops

Write a program that will calculate a few things based on sequences integers from 1 up to a given value. Your program should:

1. Ask the user to enter an integer that is greater than 0 and less than 13.
2. Use a `for` loop to sum the integers from 1 to the entered value. For example, if the user entered the number “6”, the loop should produce the sum $1 + 2 + 3 + 4 + 5 + 6 = 21$.
3. Uses a `for` loop to calculate the factorial of the entered value. For example, if if the user entered the number “6”, the loop should be produce the product, $1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 = 720$.
4. Uses a `for` loop to calculate the n-th time of the Fibonacci sequence.

Note: The Fibononacci sequence is 0, 1, 1, 2, 3, 5, 8, 13, \dots . The first items are 0 and 1, and each subsequent item of the sequence is the sum of the previous two items. ($F_n = F_{n-1} + F_{n-2}$.) For example, if the user entered the number “6”, the loop should produce the output 5. (Hint: To do this, consider using 3 variables, x, y, and z. Initialize $x = 0$ and $y = 1$. Then on each iteration, make $z = x + y$. Then move the value of y to x, and the value of z to y. Continue on for the required number of iterations.)

5. Prints the output of the for-loop summation along with the the results of the calculation: $n(n + 1)/2$, where n is the integer that was entered by the user.
6. Prints the output of the factorial calculation.
7. Prints the output of the Fibonacci calculation.

(Notes: In this particular exercise, your program must use a `for` loop, not a `while` loop. The reason for keeping the input value relatively small is that taking the factorial of a number leads to extremely large values in a hurry. Anything bigger than 12 for the input will overflow a regular 32-bit integer. Try it, to see what happens. You can extend the range by

using long long (64-bit) integers, which would allow input values as big as 20 before overflow occurs. You should try that, too. If you want to use even bigger numbers, then you will have to switch to floating point.)

C. Time

Use the functions in the time library to make a program that will pick Iowa Powerball lottery numbers for you. Your program should:

1. Use the time function to properly seed the random number generator.
2. Generate 5 random numbers between 1 and 69. These are the “white balls” in the lottery.
3. Generates one random number between 1 and 26. This is the “red power ball”.
4. Prints out the date and time that the numbers were chosen.
5. Prints out the 6 numbers, clearly indicating which is the powerball.

Note: In the real game, the numbers chosen for the white balls can be used only once. To meet the minimum requirements for this exercise, repeated numbers are allowed. As an add-on challenge, include some code so that white ball numbers can not be repeated. This is optional and is not needed to achieve a full score.

D. Quiz

Finally, there will a short quiz on while and for loops.

“Reporting”

The three programs can be written prior to lab. You can demo the programs on your own laptop brought to lab or using VirtualBox in the lab room — it’s your choice. Your instructor will first ask you to demonstrate one of the programs. A correctly functioning program will earn 15 points. The instructor will then ask you to make one or two modifications to your program. Then you will demo the modified program. If you successfully modify the program, then you will earn an extra 10 points for a total of 25 for the program. You should then print out and submit a written copy of your original program. (You can print out your original version prior to lab, if you are 100% sure that it works properly.)

The process is then repeated for the other program, which are also worth 25 points.

The quiz is worth 25 points, so that the entire lab is worth 100 points.