

Write a complete C++ program, including comments in both the main program and in each function. The program will send all output to a file. Plan: first write **sumcubes**, then **howmanyodd**, then **findsumandprod**, and finally **introduction**. This is the order in which you will be shown the topics needed.

The main program will start by calling a function named **introduction** which prints to the output file your name and a description of what the program does. **Write this function last.** . Call the function once, as the first item in the program AFTER the declaration. The **introduction** function will not be sent any parameters, and it will not return a value. It will print several lines of output explaining what the program does.

The main program will next ask the user to type in an integer, which main will store in num. Then main will call the function **sumcubes**, sending it num as a parameter value. The function **sumcubes** will find and return the sum of the first num cubes. The main program should print the value returned from the function (together with an appropriate message).

Next the main program will use a loop to read in and process 7 sets of three integer values. (Just let the loop count to 7.) Each of the integers can be positive, negative, or zero, some even and some odd. For each group of integer values, the program will do the following processing:

1. The main program will ask the user to type in three integer values, and the main program will read in and then print the three numbers.
2. The main program will call a function named **howmanyodd**, sending it the three integer values. The function will determine how many (0, 1, 2, or 3) of the three values are odd and send the answer back to the main program. The answer sent back will be a number from 0 to 3. (Hint: Odd means NOT evenly divisible by 2, and there is an operator that will tell you the remainder.)

The main program will print a message explaining the value returned. For example, if you send the values 4 5 -9 to the function, the function will return 2, and the main program will print a message like this: 2 of these numbers is/are odd.

3. The main program will then call a function named **findsumandprod** sending it two of the integers at a time. The function will determine the sum and the product of these two values and print them, together with appropriate messages. Then the main program will call the function **findsumandprod** two more times, each time sending it a different set of two values from the group of three.

To repeat: The main program will call **findsumandprod** three times:

- once with the first and second integers,
- once with the second and third integers,
- once with the third and first integers

For each call, the function will print the two integers, their sum, and their product, together with clear messages. Do NOT send all three values to the function in one call.

For example, if the numbers are 4 5 and -9, the main program should first send 4 and 5 to the function to be processed. Then the main program should send 5 and -9 to the function. Then main

should send 4 and -9 to the function. See below for sample output. This function will print, but it will not return a value to the main program.

4. Then the main program will skip a few lines and go back to step 1.

DATA: Type in 7 sets of three data values. Have one set where all three values are odd, two sets where two values are odd (and make them occur in different places (first, second, third) in the 3 sets) , a set where none are odd, and three sets where 1 is odd (and make it occur in different places (first, second, third) in the three sets). Make sure some odd values are negative.

You will be judged on the quality of your data.

STYLE: Be sure that each function has a good comment explaining two things: exactly what parameter(s) the function will receive, and exactly what the function will do. For example, if the function returns an answer, say what that answer represents; if the function prints, say what the function prints. Mention parameters by name in the comment.

OUTPUT: Send all output to a file, and have only the prompts and data entry show on the screen. Be sure to skip a few lines between groups so that everything the program prints is labeled and readable.

IMPORTANT NOTE: In order for your program to send output to a file from main AND from the functions, you must relocate the file declaration. Place it ABOVE the main program with the prototypes.

Submit the program with the subject line A3 1110.

Here is some file sample input and output (ignoring the introduction)

Input values:

8

4 5 -9

-4 -10 0

... 5 more sets of values....

Output:

The sum of the first 8 cubes is 1296

the three original integers are 4 5 -9

2 of these numbers is/are odd

4 and 5: their sum is 9 and their product is 20

5 and -9: their sum is -4 and their product is -45

4 and -9: their sum is -5 and their product is -36

the three original integers are = -4 -10 0

0 of these numbers are odd

-4 and -10: their sum is -14, their product is 40

-10 and 0: their sum is -10, their product is 0

0 and -4: their sum is -4, their product is 0

.... 5 more sets of output results...