



STUDYDADDY

**Get Homework Help
From Expert Tutor**

Get Help

C++ Programming Programming Assignment 4

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 formatting 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 Disciplinary.

Problem 1

Recursion

10 points

Implement a program that lets the user enter a positive number and then prints the result calculated by the following function. Validate the input until a positive integer is entered. Write a recursive function that computes $n * n/3 * n/9 * n/27 * n/81 \dots$ where n is a nonnegative integer, n/x is integer division and the quotient is always a multiple of 3 from the previous quotient. The sequence repeats as long as $n/x > 0$. For example:

```
27: 27 * 9 * 3 * 1 = 729
8: 8 * 2 = 16
100: 100 * 33 * 11 * 3 * 1 = 108900
```

The function prototype should be: `int recursive(int n);`

Example:

```
Input: 27
Result: 729
```

Problem 2

Lunch Box Class

10 points

Design a class named `LunchBox`. The class should have the member variables *width*, *size*, *color*, and *food*. Declare all member variables as *private*. Add a default constructor and a constructor that expects values for all member variables as parameters and assigns each value to the corresponding member variable. Also, add a getter function for each member variable. Add a setter function for the *food* member variable only. Declare all constructors and functions as *public*.

In the main, create one object of the LunchBox class. Create a second instance as a pointer to an object. Set the values for all member variables of each object.

Library: Don't forget to *include* `<string>`

Note: The main does not require any user input.

Problem 3

Home Mortgage Class

10 points

Design a class that determines the monthly home mortgage payment. The monthly payment with interest compounded monthly is calculated as follows:

$$\mathbf{payment} = \frac{\mathit{loan} \times \frac{\mathit{rate}}{12} \times \mathit{term}}{\mathit{term} - 1} \quad \mathbf{where} \quad \mathit{term} = \left(1 + \frac{\mathit{rate}}{12}\right)^{12 \times \mathit{years}}$$

Note:

payment = the monthly required payment

loan = the dollar amount of the loan

rate = the interest rate

years = the number of years of the loan

The class should have member functions for setting the loan amount, interest rate, and number of years of the loan. It should also have member functions for returning the monthly payment amount and the total amount paid to the bank at the end of the loan period. It should have a function named *calculate*, that calculates the monthly payment and stores the result in a member variable. Declare all member variables *private* and the constructor, getter and setter functions *public*.

In the main, prompt for the required input values (do not accept negative values), create an object of the home mortgage class, set the values, and output the calculated monthly payment and total amount paid to the bank.

Library: You may use the *pow* function from *include* `<cmath>` for exponentials.



STUDYDADDY

**Get Homework Help
From Expert Tutor**

Get Help