A screenshot of a computer program

AI-generated content may be incorrect.

A computer screen shot of white text

AI-generated content may be incorrect.

In this project, I worked on a C++ program that performs math operations like addition and subtraction. The goal was to make sure the program doesn't crash or give wrong answers when numbers get too big (called overflow) or too small (called underflow). To solve this, I added safety checks in the code before doing the math. For example, in the add\_numbers function, I checked if the result of adding two numbers would go over the maximum value allowed for that data type. If it would, the program throws an error and tells the user that an overflow happened. I did something similar in the subtract\_numbers function to stop underflow. I also used try-catch blocks in the testing code to catch these errors and show clear messages instead of letting the program fail. I used features from the C++ standard library, like std::numeric\_limits, to find the highest and lowest possible values for different number types. This made the solution work for different data types like int, unsigned int, or long. One challenge I had was figuring out how to make the code work for both positive and unsigned numbers, but I solved it by using simple conditions and exceptions. Overall, this project helped me understand how to safely handle math operations in programming and how to write code that can prevent serious bugs.