

**Title:** Lessons from Operator Overloading in C++

### **Answer to Problems**

Problem 1:

1. Arithmetic operator '+' is overloaded in class definition
2. The return type and argument type are both of Point

### **Notes**

Introduction: Operator overloading is a powerful feature in C++ that allows programmers to redefine the behavior of operators for user-defined data types. This report encapsulates my learnings from exploring the nuances and applications of operator overloading.

#### **User-Defined Types:**

- Operator overloading is particularly beneficial when working with user-defined types, as it allows us to apply familiar operators to these types, enhancing code readability.
- This is especially useful for creating domain-specific languages or abstractions.

#### **Global vs. Friend Functions:**

- Operator overloading can be achieved through global functions or friend functions. Friend functions have access to private members of the class.
- Choosing between global and friend functions depends on the design and encapsulation goals of the class.

#### **Conclusion:**

By carefully defining the behavior of operators for user-defined types, we can bring a level of abstraction and clarity to our programs, making them more maintainable and user-friendly. Understanding the principles of operator overloading is a valuable skill in the arsenal of a C++ developer.