Темы

* классы и объекты,
* Inline методы,
* конструкторы “по умолчанию” и “инициализации”,
* математическая библиотека cmath.

В данной работе разработан класс Complex для работы с комплексными числами, демонстрирующий ключевые аспекты C++ программирования, включая перегрузку операторов и управление памятью.

Ответы на вопросы (пп. 2 и 3 задания):

п.2: Re и Im (public double Re, Im;) хранят действительную и мнимую части. В практических сценариях лучше делать их private и предоставить доступ через геттеры/сеттеры для инкапсуляции.

п.3: Оператор ostream& operator<<(ostream& o, const Complex& c) возвращает ссылку на поток (ostream&) для поддержки цепочных операций вывода (например, cout << a << b). Использование const Complex& c предотвращает изменение объекта Complex и передает его по ссылке для повышения эффективности.

Inline-функции - это функции, код которых компилятор пытается встроить непосредственно в место вызова, минуя накладные расходы, связанные с обычным вызовом функций. Они подходят для небольших, часто вызываемых функций (геттеры/сеттеры, простые операторы), что позволяет повысить скорость работы программы.

Динамические объекты и массивы (C++):

Создание: new Complex(1, 2); - выделяет память для одного объекта и вызывает конструктор. new Complex[3]; - выделяет память для массива объектов (вызывает конструктор по умолчанию для каждого элемента).

Уничтожение: delete pc; - освобождает память для одного объекта. delete[] pcArray; - освобождает память для массива объектов. Важно использовать delete[] для массивов, иначе поведение не определено!

Корректное использование new и delete (или new[] и delete[]) для динамической памяти предотвращает утечки памяти, обеспечивая надежность программы.