

Выполнил(а) Лагус М.С., № группы 3114, оценка _____
Фамилия И.О. студента не заполнять

Название статьи/главы книги/видеолекции		
Шаблон программирования Pimpl - то, что вам следует знать		
ФИО автора статьи (или e-mail)	Дата публикации (не старше 2018 года)	Размер статьи (от 400 слов)
Александр Кузьминых	" 15 " 01 2018 г.	2214
Прямая полная ссылка на источник и сокращённая ссылка (bit.ly, goo.gl, tr.im и т.п.)		
https://evileg.com/ru/post/322/ https://bit.ly/3oKQsfl		
Теги, ключевые слова или словосочетания		
Шаблон программирования, идиома, оптимизация времени компиляции, современный C++, брандмауэр компиляции		
Перечень фактов, упомянутых в статье		
1. При изменении даже скрытых деталей реализации класса, необходимо перекомпилировать все файлы, содержащие объекты данного класса 2. Использование указателя на объект класса не требует наличия реализации класса в зоне видимости указателя 3. Вынесение скрытой реализации в отдельный класс может изменить работу с динамическим выделением памяти, поменяв область видимости реализации класса		
Позитивные следствия и/или достоинства описанной в статье технологии (минимум три пункта)		
1. Применение данной идиомы позволяет существенно сокращать время компиляции крупных проектов 2. Вынесение приватных полей из класса не противоречит принципу инкапсуляции данных, даже в какой-то степени помогая сокрытию деталей реализации 3. Данный шаблон программирования можно адаптировать к текущим стандартам написания кода на C++, используя в его имплементации другие популярные идиомы		
Негативные следствия и/или недостатки описанной в статье технологии (минимум три пункта)		
1. Данный шаблон не так сильно известен, что может привести к недопониманию других программистов, увидевших такой код 2. Усложняется поддержка и отладка программы, класс разделён на части 3. Ухудшается производительность, добавляется дополнительный уровень косвенного обращения внутри исходного класса		
Ваши замечания, пожелания преподавателю или анекдот о программистах¹		
Данный шаблон заинтересовал меня тем, что он направлен на улучшение времени компиляции, ценой небольшой потери производительности в runtime, тогда как многие другие инструменты языка позволяют действовать наоборот, выносить часть вычислений с времени выполнения на время компиляции. Это показывает, что под разные требования доступны разные инструменты, порой даже решающие противоположные по своей идее задачи.		

¹ Наличие этой графы не влияет на оценку