

Школа:		
Дата:	ФИО учителя:	
Класс:	Участвовали:	Не участвовали:
Тема урока: Классы стандартной библиотеки C++.		
Цели обучения, которые достигаются на данном уроке	Записывать алгоритмы на языке программирования с помощью стандартной библиотеки	
Воспитательности	<ul style="list-style-type: none"> • Обучение на протяжении всей жизни; • Уважение; • Сотрудничество. 	
Предварительные знания	Особенности объектно-ориентированного программирования на C++.	
Межпредметные связи	информатика	
Запланированные этапы урока	Запланированная деятельность на уроке	Ресурсы
Начало урока _7_ мин	<p>Организационный момент: проверка присутствующих по журналу.</p> <p>Ознакомление учащихся с темой урока, его целями и критериями успеха.</p> <p>Актуализация знаний, повторение «Особенности объектно-ориентированного программирования на C++», через решение жизненной ситуации:</p>	Презентация слайды 1-4
Середина урока _25_ мин	<p>Стандартная библиотека C++ (далее – STD) представляет собой большой набор функций и классов для решения различных задач. STD и ядро языка C++ развивались и стандартизировались параллельно. В следствие этого, многие важные вещи, например ввод/вывод, не являются частью ядра, а принадлежат STD. Все компоненты STD согласованы друг с другом, но основаны на разных принципах проектирования. В основу разработки библиотеки положена парадигма обобщенного программирования. Все классы библиотеки – это шаблоны.</p> <p>Первоначально Библиотека стандартных шаблонов (англ. Standard Template Library, STL) выделялась как обособленная часть STD. Но, в настоящее время, STL полностью интегрирована в STD и рассматривается как её неотъемлемая часть (хотя в некоторых источниках по прежнему фигурирует аббревиатура STL, и сама эта библиотека рассматривается как некая часть STD, но это не соответствует действительности). Обобщенная парадигма распространилась и на остальные части STD, первоначально не входившие в STL. Архитектуру STL разработали Александр Степанов и Менг Ли. STL вошла в стандартную библиотеку C++ весной 1994 года.</p> <p>STD также включает в себя спецификации стандарта ISO C11 (в версии стандарта C++17) стандартной библиотеки языка Си. Каждый заголовочный файл из стандартной библиотеки языка Си включен в STD под именами, начинающиеся префиксом 'c', например, "ctime".</p> <p>По правилам языка C++ соответствующий компонент STD</p>	

	<p>добавляется в программу путем включения заголовочного файла с помощью директивы:</p> <pre>#include <имя_файла></pre> <p>при этом обычное “расширение” в имени заголовочных файлов (".h") отбрасывается.</p> <p><i>Примечание.</i> На первом уроке в 10 классе мы акцентировали ваше внимание на том, что полной совместимости между языками C и C++ не существует. В современном стандарте языка C++ появилось много вещей, которые не будут работать в языке Си, равно как и многие вещи принятые в Си перестали работать в C++. Например (далеко неполный список):</p> <pre>typedef int name; // В C++ не работает</pre> <pre>char array[4] = "abcd"; // В C++ не работает</pre> <pre>char* p = "abc"; // В C++ не работает</pre> <pre>sizeof(A) == sizeof(int) // работает в C, но не в C++</pre> <pre>sizeof(A) == sizeof(e) // работает в C ++, но не в C</pre> <pre>void f() { auto int x; // в C++ не работает }</pre> <pre>static struct S { int i; // в C++ не работает };</pre> <p>Таким образом, использовать библиотеки языка Си вы можете, но писать совместимый код на современном C++ не получится. В языке Си не существуют классы динамических контейнеров, которые освобождают память в автоматическом режиме. Динамические массивы в Си нужно создавать “вручную” с помощью библиотечных функций malloc, free и realloc. (В C++ поддерживаются функции работы с памятью из стандартной библиотеки Си, но их использование не рекомендуется). В продолжении нашего курса мы практически не обращаемся к библиотеке языка Си (за исключением smath и stime).</p> <p>Физминутка.</p> <p>(I) Работа над проектом Ученики используют составные условия при разработке собственных проектов.</p>	
<p>Конец урока _8_ мин</p>	<p>Провожу рефлексию в виде ответов/ пожеланий/ вопросов на стикерах.</p>	<p>Рефлек сия, маркеры</p>

Дифференциация – каким образом Вы планируете оказать больше поддержки? Какие задачи Вы планируете поставить перед более способными учащимися?	Оценивание – как Вы планируете проверить уровень усвоения материала учащимися?	Охрана здоровья и соблюдение техники безопасности
<p>Индивидуальная работа и работа всем классом.</p>	<p>Использовались методы наблюдения, самооценки, проверки учителем</p>	<p>Правила техники безопасности соблюдались полностью. Физминутка.</p>