

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный университет имени М.В. Ломоносова»  
Химический факультет

УТВЕРЖДАЮ

Декан химического факультета,  
Акад. РАН, профессор



/В.В. Лунин/

«27» февраля 2017 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**  
**Объектно-ориентрованное программирование в C++**

**Уровень высшего образования:**  
Специалитет

---

**Направление подготовки (специальность):**  
04.05.01 Фундаментальная и прикладная химия

**Направленность (профиль) ОПОП:**  
Физическая химия

**Форма обучения:**  
очная

---

Рабочая программа рассмотрена и одобрена  
Учебно-методической комиссией факультета  
(протокол №1 от 27.01.2017)

Москва 2017

Рабочая программа дисциплины разработана в соответствии с самостоятельно установленным МГУ образовательным стандартом (ОС МГУ) для реализуемых основных профессиональных образовательных программ высшего образования по направлению подготовки / специальности 04.05.01 «Фундаментальная и прикладная химия» (программа специалитета), утвержденного приказом МГУ от 22 июля 2011 года № 729 (в редакции приказов МГУ от 22 ноября 2011 года № 1066, от 21 декабря 2011 года № 1228, от 30 декабря 2011 года № 1289, от 27 апреля 2012 года № 303, от 30 декабря 2016 года № 1671).

Год (годы) приема на обучение

2014/2015, 2015/2016, 2016/2017, 2017/2018, 2018/2019

---

1. Наименование дисциплины (модуля) **Объектно-ориентрованное программирование в С++.**
2. Уровень высшего образования – **специалитет.**
3. Направление подготовки: **04.05.01 Фундаментальная и прикладная химия.**
4. Место дисциплины (модуля) в структуре ООП: вариативная часть ООП, блок ПД.
5. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)

Компетенция	Планируемые результаты обучения по дисциплине (модулю)
<b>СПК-4.С.</b> Способность использовать физические и математические модели с учетом их возможностей и ограничений при обработке и интерпретации экспериментальных данных в избранной области физической химии	<b>Знать:</b> основные концепции объектно-ориентированного программирования; <b>Уметь:</b> проектировать и создавать программы, реализующие подходы ООП для решения различных задач; <b>Владеть:</b> навыками работы с компиляторами g++ и icc, средами разработки и отладки программ; <b>Владеть:</b> навыками использования специализированных библиотек линейной алгебры и создании make-файлов.

6. Объем дисциплины в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся:

объем дисциплины (модуля) составляет 3 зачетных единиц, всего 108 часов, из которых 40 часов составляет контактная работа студента с преподавателем (18 часов – занятия лекционного типа, 18 часов – занятия семинарского типа, 2 часа – групповые консультации, 2 часа – промежуточный контроль успеваемости), 68 часов составляет самостоятельная работа учащегося.

7. Входные требования для освоения дисциплины (модуля), предварительные условия.

Обучающийся должен

**Знать:** основные этапы выполнения программного кода процессором; общие принципы построения алгоритмов.

**Уметь:** работать в консольном режиме на удаленных серверах; редактировать и изменять атрибуты файлов; работать с потоками ввода вывода.

**Владеть:** минимальными навыками написания программ на любом языке; основными возможностями редактора Vim.

8. Содержание дисциплины (модуля), структурированное по темам.

Наименование и краткое содержание разделов и тем дисциплины (модуля),  форма промежуточной аттестации по дисциплине (модулю)	Всего (часы)	В том числе								
		Контактная работа (работа во взаимодействии с преподавателем), часы из них					Самостоятельная работа обучающегося, часы из них			
		Занятия лекционного типа	Занятия семинарского типа	Групповые консультации	Индивидуальные консультации	Учебные занятия, направленные на проведение текущего контроля успеваемости, промежуточной аттестации	Всего	Выполнение домашних заданий	Подготовка рефератов и т.п..	Всего

Тема 1. Функция main(); простые типы данных; области видимости переменных; потоки ввода/вывода; поток ошибок; понятие блока программы; циклы (for, while); функции и их сигнатура; реализация булевой алгебры, директивы компилятора.	<b>6</b>	1	1				<b>2</b>	4		<b>4</b>
Тема 2. Подключение стандартных библиотек; пространство имен; операции ввода/вывода в файл; манипуляторы потока; обзор библиотеки <cmath>; одномерные массивы	<b>7</b>	1	1				<b>2</b>	5		<b>5</b>
Тема 3. Указатели, ссылки, адресная арифметика. Реализация многомерных массивов. Особенности передачи массивов и указателей в функцию. С-строки. Работа с аргументами командной строки.	<b>7</b>	1	1				<b>2</b>	5		<b>5</b>
Тема 4. Составные типы данных. Классы. Оформление классов в отдельные файлы. Заголовочные файлы. Сторожа. Дружественные функции. Понятие инкапсуляции. Перегрузка операторов	<b>7</b>	1	1				<b>2</b>	5		<b>5</b>
Тема 5. Наследование и его типы. Виртуальные классы. Полиморфизм.	<b>8</b>	2	1				<b>3</b>	5		<b>5</b>

Тема 6. UML диаграммы. Паттерны проектирования. Паттерны «Одиночка» и «Мост»	<b>9</b>	2	2				<b>4</b>	5		<b>5</b>
Тема 7. Паттерны «Абстрактная фабрика», «MVC», «Наблюдатель»	<b>8</b>	2	2				<b>4</b>	4		<b>4</b>
Тема 8. Анализ паттернов ООП.	<b>8</b>							8		<b>8</b>
Тема 9. Понятие контейнера и итератора. Обзор основных контейнеров библиотеки STL.	<b>6</b>	1	1				<b>2</b>	4		<b>4</b>
Тема 10. Детали реализации некоторых контейнеров STL. Красно-черные деревья. Хэш-функции.	<b>8</b>	2	2				<b>4</b>	4		<b>4</b>
Тема 11. Составление make-файлов. Обзор систем контроля версий, git.	<b>9</b>	2	2				<b>4</b>	5		<b>5</b>
Тема 12. Отладка программ. Среды разработки.	<b>7</b>	1	1				<b>2</b>	5		<b>5</b>
Тема 13. Обзор библиотеки LAPACK. Использование функций языка Fortran в программе на C++. Изучение функционала библиотеки Lapack	<b>7</b>	1	1				<b>2</b>	5		<b>5</b>
Тема 14. Понятие контейнера и итератора. Обзор основных контейнеров библиотеки STL.	<b>9</b>	1	2	2			<b>5</b>	4		<b>4</b>

Промежуточная аттестация: <i>зачет</i>	2					2	2			
<b>Итого</b>	<b>108</b>	<b>18</b>	<b>18</b>	2		2	<b>40</b>	68		<b>68</b>

### 9. Образовательные технологии:

- применение компьютерных симуляторов, обработка данных на компьютерах, использование компьютерных программ, управляющих приборами;
- использование средств дистанционного сопровождения учебного процесса;
- преподавание дисциплин в форме авторских курсов по программам, составленным на основе результатов исследований научных школ МГУ.

### 10. Учебно-методические материалы для самостоятельной работы по дисциплине (модулю):

Студентам предоставляется программа курса, план занятий и, по ходу занятий, – наборы заданий для самостоятельной работы. По теме каждого занятия указывается материал в источниках из списков основной и вспомогательной литературы.

### 11. Ресурсное обеспечение:

Перечень основной и вспомогательной учебной литературы ко всему курсу

### Основная литература

Б. Страуструп. Язык программирования C++. Изд-во: Бинум, 2015, -1136с.

### Дополнительная литература

1. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. Серия Библиотека программиста. Изд-во: Питер. 2016, -366 с.

2. [www.cplusplus.com](http://www.cplusplus.com)

12. Язык преподавания – русский

### 13. Преподаватели:

доц., к.ф.-м.н., Безруков Дмитрий Сергеевич, кафедра физической химии химического факультета МГУ;

e-mail: dsbezrukov@gmail.com

### **Фонды оценочных средств, необходимые для оценки результатов обучения**

Образцы оценочных средств для текущего контроля усвоения материала и промежуточной аттестации - зачета. На зачете проверяется достижение промежуточных индикаторов компетенций, перечисленных в п.5.

#### **Вопросы для зачета**

1. Типы данных в C++. Статическое и динамическое приведение типов.
2. Стандартные потоки. Библиотека <iostream>. Манипуляторы потоков.
3. Режимы работы компилятора. Директивы компилятора. Этапы компиляции программы.
4. Стандартные библиотеки.
5. Области видимости. Пространство имен.
6. Библиотека <cmath>.
7. Устройство памяти. Адресная арифметика. Указатели и ссылки.
8. Передача аргумента по значению и по ссылке.
9. Строковые переменные. Класс <string> и C-строки.
10. Понятие составного типа переменных. Структуры и классы.
11. Методы, поля, свойства.
12. Наследование и его типы.
13. Перегрузка компиляторов.
14. UML-диаграммы.
15. Основная классификация паттернов проектирования.
16. Контейнеры — общая классификация.
17. Устройство контейнера <map> и <vector>.
18. Make-файлы. Синтаксис. «Цели» в make-файле.
19. Подключение сторонних библиотек. Внешние функции.
20. Библиотека Lpack. Устройство и основной функционал.

#### **Методические материалы для проведения процедур оценивания результатов обучения**



Шкала оценивания знаний, умений и навыков является единой для всех дисциплин (приведена в таблице ниже)

<b>ШКАЛА И КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ по дисциплине (модулю)</b>				
Оценка	2	3	4	5
Результат				
Знания	Отсутствие знаний	Фрагментарные знания	Общие, но не структурированные знания	Сформированные систематические знания
Умения	Отсутствие умений	В целом успешное, но не систематическое умение	В целом успешное, но содержащее отдельные пробелы умение (допускает неточности непринципиального характера)	Успешное и систематическое умение
Навыки (владения)	Отсутствие навыков	Наличие отдельных навыков	В целом, сформированные навыки, но не в активной форме	Сформированные навыки, применяемые при решении задач

<b>РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине (модулю)</b>	<b>ФОРМА ОЦЕНИВАНИЯ</b>
<b>Знать:</b> основные концепции объектно-ориентированного программирования;	мероприятия текущего контроля успеваемости, устный опрос на зачете
<b>Уметь:</b> проектировать и создавать программы, реализующие подходы ООП для решения различных задач	мероприятия текущего контроля успеваемости, устный опрос на зачете
<b>Владеть:</b> навыками работы с компиляторами g++ и icc, средами разработки и отладки программ <b>Владеть:</b> навыками использования специализированных библиотек линейной алгебры и создании make-файлов	мероприятия текущего контроля успеваемости, устный опрос на зачете