

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор физтех-школы
прикладной математики и
информатики**

А.М. Райгородский

по дисциплине:	Рабочая программа дисциплины (модуля)
по направлению:	Объектно-ориентированное программирование на C++
профиль подготовки:	Информатика и вычислительная техника
курс:	Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
квалификация:	2 бакалавр

Семестры, формы промежуточной аттестации:

3 (осенний) - Дифференцированный зачет

4 (весенний) - Дифференцированный зачет

Аудиторных часов: 240 всего, в том числе:

лекции: 0 час.

семинары: 120 час.

лабораторные занятия: 120 час.

Самостоятельная работа: 210 час.

Всего часов: 450, всего зач. ед.: 10

Программу составил: В.В. Яковлев, канд. физ.-мат. наук

Программа обсуждена на заседании кафедры алгоритмов и технологий программирования 26.08.2021

Аннотация

Дисциплина направлена на освоение концепций объектно-ориентированного программирования на примере языка C++. В ходе изучения будут рассмотрены основные такие понятия объектно-ориентированного программирования, как инкапсуляция данных, наследование классов, и использование шаблонов. Кроме того, будут разобраны практические примеры проектирования программ и изучено устройство стандартной библиотеки C++.

1. Цели и задачи

Цель дисциплины

овладение студентами языком программирования C++ и приемами использования этого языка в практике программирования

Задачи дисциплины

- приобретение студентами навыков проектирования и реализации программ на языке C++
- овладение студентами приемами объектно-ориентированного программирования на языке C++ с использованием стандартной библиотеки
- овладение навыками работы со стандартными инструментами разработки для C++.

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны знать:

концепции объектно-ориентированного программирования

уметь:

программировать с использованием объектов и классов

владеть:

навыками декомпозиции программ на компоненты

4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа

1	Раздел 1. Основы объектно-ориентированного программирования		20	20	35
2	Раздел 2. Проектирование и разработка классов на языке C++		40	40	70
3	Раздел 3. Проектирование программ на языке C++ и реализация нескольких классов		25	25	50
4	Раздел 4. Шаблоны и основы метапрограммирования		35	35	55
Итого часов			120	120	210
Подготовка к экзамену		0 час.			
Общая трудоёмкость		450 час., 10 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 3 (Осенний)

1. Раздел 1. Основы объектно-ориентированного программирования

Парадигмы программирования: процедурное, модульное, объектно-ориентированное. Абстракции данных

Классы и члены классов. Приватные, защищенные и открытые поля класса. Инкапсуляция данных.

Конструкторы и деструктор класса. Перегрузка операторов. Реализация конструкторов копирования и присваивания.

2. Раздел 2. Проектирование и разработка классов на языке C++

Константные аргументы, поля и методы класса.

Ссылочные типы данных, их отличия от указателей.

Статические поля и методы класса. Синглтоны и фабричные методы создания объектов класса.

Наследование классов. Множественное наследование.

Виртуальные методы. Чистые виртуальные методы и абстрактные классы. Интерфейсы.

Семестр: 4 (Весенний)

3. Раздел 3. Проектирование программ на языке C++ и реализация нескольких классов

Приведение типов в C++. Статическое, динамическое и константное приведение типов.

Низкоуровневое представление типов данных. Приведение типа по указателю без проверок (reinterpret_cast). Тип объединения.

Декомпозиция программы на составные модули. Раздельная компиляция. Заголовочные файлы.

4. Раздел 4. Шаблоны и основы метапрограммирования

Шаблоны. Реализация шаблонных функций и классов.

Итераторы. Категории итераторов в C++. Использование и реализация итераторов.

Семантика перемещения объектов. Функция std::move и конструктор перемещения.

Обзор стандартной библиотеки C++ STL.

5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

- для лекционных занятий: учебная аудитория оборудованная с мультимедийным проектором и экраном или широкоформатным большим монитором для осуществления презентации лекционных материалов, ноутбук или стационарный компьютер, стулья, парты, стол преподавателя, доска;
- для лабораторных занятий: стулья, парты для учащихся, стол преподавателя, доска. аудиторная, трибуна, видеопроектор, мультимедийный проектор с экраном или монитор, ноутбук;
- для самостоятельной работы обучающегося: компьютеры с возможностью подключения к сети «Интернет» и обеспечены доступом в электронную информационно-образовательную среду ВУЗа, точка доступа wi-fi, для возможности использования обучающимися персональных ноутбуков, доска аудиторная, стол и стул для преподавателя, столы и стулья для студентов, трибуна.

6.Перечень рекомендуемой литературы

Основная литература

1. Язык программирования С++ [Текст] / Б. Страуструп ; пер. с англ. С. Анисимова, М. Кононова ; под ред. Ф. Андреева, А. Ушакова .— Спец. изд. с авт. изменениями и доп. — М. : Бином Пресс, 2008 .— 1104 с.

Дополнительная литература

1. Стандарты программирования на С++ [Текст], 101 правило и рекомендация/Г. Саттер, А. Александреску, -М.; СПб; Киев, Изд. дом "Вильямс", 2015

7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Не используются

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)

Информационная система «Единое окно доступа к образовательным ресурсам» (ИС «Единое окно») <http://window.edu.ru/> Свободный доступ

Платформа открытое образование <https://openedu.ru/> Свободный доступ

Microsoft Visual Studio

9. Методические указания для обучающихся по освоению дисциплины (модуля)

Успешное освоение курса требует напряжённой самостоятельной работы студента. В программе курса приведено минимально необходимое время для работы студента над темой. Самостоятельная работа включает в себя:

- проработку учебного материала (по конспектам лекций, учебной и научной литературе), подготовку ответов на вопросы, предназначенных для самостоятельного изучения.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению:	Информатика и вычислительная техника
профиль подготовки:	Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
курс:	2
квалификация:	бакалавр
Семестры, формы промежуточной аттестации:	
	3 (осенний) - Дифференцированный зачет
	4 (весенний) - Дифференцированный зачет
Разработчик:	В.В. Яковлев, канд. физ.-мат. наук

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации

2. Показатели оценивания компетенций

В результате изучения дисциплины «Объектно-ориентированное программирование на C++» обучающийся должен:

знать:

концепции объектно-ориентированного программирования

уметь:

программировать с использованием объектов и классов

владеть:

навыками декомпозиции программ на компоненты

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

С целью контроля освоения обучающимися учебного материала проводится устный опрос в начале занятия по теме прошлого занятия.

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

Примерный перечень заданий:

Перечень заданий для текущего контроля успеваемости:

Задание 1. Реализация класса с конструкторами

Реализуйте класс Stack, реализующий структуру стека для целочисленных элементов. В классе необходимо реализовать основные методы добавления и извлечения элементов, конструктор и деструктор.

Задание 2. Перегрузка операторов, константные методы и передача аргументов по ссылке

Реализуйте класс String, который является представлением строки символов. Для класса реализуйте все возможные конструкторы, деструктор, операции сравнения строк, и доступ к отдельным символам строки.

Задание 3. Реализация классов с иерархией наследования

Реализуйте иерархию классов для представления геометрических примитивов (окружность, квадрат, треугольник) на плоскости, и реализуйте функцию, которая определяет, пересекаются ли два произвольных объекта.

Задание 4. Шаблонные классы

Реализуйте класс односвязного списка над произвольным типом данных с помощью шаблона. В реализации необходимо предусмотреть всевозможные конструкторы, включая конструктор перемещения, операторы сравнения, а также реализовать прямой итератор и реверсивный итератор.

Пример билета:

1. Реализуйте класс `Stack`, реализующий структуру стека для целочисленных элементов. В классе необходимо реализовать основные методы добавления и извлечения элементов, конструктор и деструктор.
2. Реализуйте класс `String`, который является представлением строки символов. Для класса реализуйте все возможные конструкторы, деструктор, операции сравнения строк, и доступ к отдельным символам строки.

Критерии оценивания

отлично

- 10 Полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы, код оформлен в едином удобочитаемом стиле
- 9 Полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач, реализованы оптимальные алгоритмы
- 8 Полностью и вовремя решены все задачи без ошибок. Продемонстрирован грамотный подход к решению задач

хорошо

- 7 Полностью решены все задачи. Допущены несущественные ошибки.
- 6 Полностью решено большинство задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.
- 5 Полностью решено две трети задач. В некоторых задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.

удовлетворительно

- 4 Полностью решено более половины задач. В остальных задачах допущены и не исправлены ошибки, либо некоторые задачи решены частично.
- 3 Полностью решено более половины задач.

неудовлетворительно

- 2 Решено менее половины задач.
- 1 Не решено ни одной задачи.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Во время проведения дифференцированного зачета обучающиеся могут пользоваться программой дисциплины,