

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«Национальный исследовательский ядерный университет «МИФИ»  
**Саровий физико-технический институт –**  
филиал федерального государственного автономного образовательного учреждения высшего образования  
«Национальный исследовательский ядерный университет «МИФИ»  
**(САРФТИ НИЯУ МИФИ)**

ОДОБРЕНО  
Ученым советом, протокол №\_\_  
от «\_\_» \_\_\_\_\_ 2024 г.

## **РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

### **«Объектно-ориентированное программирование»**

**Специальность:** 09.02.07 Информационные системы и программирование

**Наименование образовательной программы:** Информационные системы и программирование

**Уровень образования:** среднее профессиональное образование

**Форма обучения:** очная

**Квалификация:** программист

**Саров, 2024**

## СОДЕРЖАНИЕ

|  |           |
|--|-----------|
| 1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ              | стр.<br>4 |
| 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ                 | 6         |
| 3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ                     | 13        |
| 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ | 14        |

# 1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

## 1.1. Область применения рабочей программы

Рабочая программа учебной дисциплины является частью основной профессиональной образовательной программы, включенной за счет вариативной части ФГОС по специальностям СПО: 09.02.03 Программирование в компьютерных системах.

## 1.2. Место учебной дисциплины в структуре основной профессиональной образовательной программы:

Учебная дисциплина Объектно-ориентированное программирование относится к циклу общепрофессиональных дисциплин профессионального цикла

## 1.3. Цели и задачи учебной дисциплины – требования к результатам освоения учебной дисциплины:

В результате освоения учебной дисциплины обучающийся должен обладать **общими компетенциями**, включающими в себя способность:

ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.

ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие.

ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами.

ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста.

ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей, применять стандарты антикоррупционного поведения.

ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.

ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности.

ОК 09. Использовать информационные технологии в профессиональной деятельности.

ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языках.

ОК 11. Использовать знания по финансовой грамотности, планировать предпринимательскую деятельность в профессиональной сфере.

### **Учащийся должен обладать профессиональными компетенциями,**

соответствующими основным видам профессиональной деятельности:

ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием.

ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4. Выполнять тестирование программных модулей.

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.

ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

В результате освоения дисциплины обучающийся должен **уметь**:

- Проектировать информационные системы на языке C++
- Проектировать и создавать программные системы Java

В результате освоения учебной дисциплины обучающийся должен **знать** :

- основ технологии объектно-ориентированной декомпозиции программных систем, базовых шаблонов проектирования (Наблюдатель, Итератор, Одиночка, Фабрика, Заместитель), отношений между классами и основ UML (диаграммы классов и последовательностей).
- особенности построения объектно-ориентированных программных систем на C++.
- основные инструментальные средства языка C++ и стандартной библиотеки
- базовые знания платформы Java, особенности построения программных систем Java
- средства реализации принципов ООП и инструментальные средства языка Java.
- основы технологий построения простейших распределенных информационных систем и обеспечения безопасности.

#### **1.4. Количество часов на освоение программы учебной дисциплины:**

максимальной учебной нагрузки студента 147 часа, в том числе:

- обязательной аудиторной учебной нагрузки обучающегося 98 часов;
- обязательной аудиторной практической работы обучающегося 62 часов;
- самостоятельной работы обучающегося 49 часов.

## 2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

### 2.1. Объем учебной дисциплины и виды учебной работы

| <b>Вид учебной работы</b>                                       | <b><i>Объем часов</i></b> |
|---|---------------------------|
| <b>Максимальная учебная нагрузка (всего)</b>                    | <b><i>168</i></b>         |
| <b>Обязательная аудиторная учебная нагрузка (всего)</b>         | <b><i>120</i></b>         |
| Лекции  | <b><i>32/16</i></b>       |
| практические занятия  | <b><i>32/32</i></b>       |
| <b>Самостоятельная работа обучающегося (всего)</b>              | <b><i>48</i></b>          |
| <b><i>Итоговая аттестация в форме экзамена в 4 семестре</i></b> |                           |

## 2.2 Тематический план и содержание учебной дисциплины Объектно-ориентированное программирование

| Наименование разделов и тем   | Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся   | Объем часов | Уровень освоения |
|---|---|-------------|------------------|
| <i>1</i>  | <i>2</i>  | <i>3</i>    | <i>4</i>         |
| <b>Раздел 1. Основные принципы объектно-ориентированного программирования</b> |   | <b>42</b>   |                  |
| <b>Тема 1.1. Эволюция методологий программирования</b>                        | <p><b>Содержание учебного материала</b><br/>                     Основное содержание дисциплины «Основы программирования». Алфавит языка Паскаль. Начало начал, или первое поколение языков программирования.<br/>                     Развитие алгоритмических абстракций, или второе поколение языков программирования. Модуль как единица построения программных систем, третье поколение языков программирования. Зарождение объектной модели, четвертое поколение языков программирования. Объектные языки программирования, объектно-ориентированные языки программирования, объектно-ориентированный анализ, дизайн и проектирование. Парадигмы программирования</p> | 6           | 1                |
|   | <p><b>Самостоятельная работа</b><br/>                     Подготовка конспекта изученного материала</p>   | 8           | 3                |
| <b>Тема 1.2. Составные части объектного подхода</b>                           | <p><b>Содержание учебного материала</b><br/>                     Абстрагирование Инкапсуляция Модульность Иерархия Типизация Параллелизм Сохраняемость приложений.</p>  | 10          | 1                |

|  |  |           |   |
|--|--|-----------|---|
|  | <p><b>Самостоятельная работа</b><br/>Темы рефератов:</p> <ol style="list-style-type: none"> <li>1. Основные принципы объектного подхода. Абстрагирование.</li> <li>2. Основные принципы объектного подхода. Инкапсуляция.</li> <li>3. Основные принципы объектного подхода. Модульность.</li> <li>4. Основные принципы объектного подхода. Иерархия.</li> </ol>                | 8         | 3 |
|  | <p><b>Практические занятия</b><br/>Работа с данными разных типов, запись арифметических выражений на языке Паскаль.<br/>Составление линейной программы<br/>Создание графического изображения с помощью стандартных процедур.<br/>Операторы для безусловных и условных переходов<br/>Оператор выбора.<br/>Составление программы с операторами условия, с операторами выбора</p> | 10        | 2 |
| <b>Раздел 2. Объектно-ориентированная модель</b>     |  | <b>49</b> |   |
| <b>Тема 2.1.<br/>Объектно-ориентированная модель</b> | <p>Объектно-ориентированная модель<br/>Понятие объекта. Свойства, присущие объектам. Состояние.<br/>Поведение. Идентичность. Отношения между объектами. Типы отношений. Связь (ассоциация). Агрегация</p>  | 2         | 1 |
|  | <p><b>Практические занятия</b><br/>Раздельная компиляция и пространства имен<br/>Перегрузка функций, указатели на функции, перечисления</p>  | 12        | 2 |
|  | <p><b>Самостоятельная работа</b><br/>Основные принципы объектного подхода. Типизация.<br/>Основные принципы объектного подхода. Параллелизм.<br/>Сохраняемость.</p>  | 8         | 3 |

|   |  |           |   |
|---|--|-----------|---|
| <b>Тема 2.2.<br/>Классы</b>                                     | <b>Содержание учебного материала</b><br>Природа классов. UML – унифицированный язык моделирования. Четырехуровневая метамодель MOF Отношения между классами. Типы отношений. Ассоциация. Агрегация Использование. Наследование. Инстанцирование. Отношения между классами и объектами  | 2         | 1 |
|   | <b>Практические занятия</b><br>Классы. Реализация упрощенного григорианского календаря<br>Классы. Перегрузка операций. Реализация матрицы и вектора<br>Иерархии классов, наследование. Реализация командного процессора  | 16        | 2 |
|   | <b>Самостоятельная работа</b><br>Классы. Природа классов. Метамодель. Инстанцирование.<br>Классы. Структура класса. Абстрактные классы и интерфейсы.<br>Классы. Отношения между классами. Ассоциация и агрегация.  | 9         | 3 |
| <b>Раздел 3. Средства объектного программирования языка C++</b> |  | <b>56</b> |   |
| <b>Тема 3.1.<br/>Основные алгоритмические отличия C++</b>       | <b>Содержание учебного материала</b><br>Использование ссылок. Передача аргументов функции по ссылке. Использование констант. Логические тип и перечисления. Операторы управления динамической памятью, инициализация массивов. Структура программы, отдельная компиляция и особенности использования статической памяти. Пространства имен и исключения (краткий обзор). Библиотека ввода вывода (краткий обзор iostream). Функциональный полиморфизм. | 4         | 1 |

|  |  |   |   |
|--|--|---|---|
|  | <p><b>Самостоятельная работа</b><br/>         Модель памяти и структура программы. Классы памяти. Ссылки.<br/>         Средства абстракции C++. Структура класса. Статические члены.<br/>         Средства инкапсуляции C++. Инкапсуляция и наследование. Друзья</p>   | 6 | 3 |
|  | <p><b>Практические занятия</b></p>   | 4 | 2 |
| <p><b>Тема 3.2.</b><br/> <b>Средства объектного программирования языка C++</b></p> | <p><b>Содержание учебного материала</b><br/>         Представление объектов и классов. Реализация поведения объектов на примере добавления функций—членов в структуры. Структура как вырожденный класс. Структура объявления класса. Доступ к членам класса. Поля данных класса как механизм реализации состояния объекта.<br/>         Статические поля и методы классов. Инициализация статических полей. Реализация отношений между объектами и классами<br/>         Ассоциация и агрегация объектов и классов. Зависимость по времени жизни. Использование и зависимость от интерфейсов. Объекты при передаче параметров и возврате из методов. Варианты реализации отношения клиент-сервер. Внутренние классы.</p> | 6 | 1 |
|  | <p><b>Самостоятельная работа</b><br/>         Объект с точки зрения ООП. Состояние. Поведение.<br/>         Объект с точки зрения ООП. Идентичность и жизненный цикл объектов.<br/>         Объект с точки зрения ООП. Взаимоотношения между объектами.</p>  | 4 | 3 |

|  |  |            |   |
|--|--|------------|---|
|  | <b>Практические занятия</b>  | 8          |   |
| <b>Тема 3.3 Наследование как средство организации иерархий классов</b> | <b>Содержание учебного материала</b><br>Наследование как средство организации иерархий классов. Принцип замещения Лисковской. Одиночное наследование. Понятие производного класса. Управление доступом в производных классах. Конструкторы и деструкторы, совмещение имен методов при наследовании, иерархии.<br>Абстрактные классы и виртуальные функции. Виртуальный полиморфизм. Информация о типе на этапе выполнения. RTTI. | 6          | 1 |
|  | <b>Практические занятия</b><br>C++: средства реализации состояния объектов; реализация поведения.<br>Перегрузка операторов.<br>Жизненный цикл объекта. Инициализация массивов. Конструкторы и деструкторы. Порядок вызова конструкторов и деструкторов при наследовании..  | 10         | 2 |
|  | <b>Самостоятельная работа</b><br>Варианты реализации отношения клиент-сервер. Объекты при передаче параметров и возврате из методов.<br>Исключения в C++. Обработка исключений.<br>Шаблоны классов и шаблоны функций. Специализация.<br>Основы STL. Структура и назначение. Контейнеры. Алгоритмы<br>Стандартная библиотека, ввод-вывод  | 8          | 3 |
|  | <b>Всего</b>   | <b>147</b> |   |

Для характеристики уровня освоения учебного материала используются следующие обозначения:

- ознакомительный (узнавание ранее изученных объектов, свойств);
- репродуктивный (выполнение деятельности по образцу, инструкции или под руководством)
- продуктивный (планирование и самостоятельное выполнение деятельности, решение проблемных задач)

### 3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

Технические средства обучения и контроля:

1. Лекционные залы с проекторами.
2. Показ слайдов и презентаций.
3. Персональные ЭВМ.
4. Электронные эквиваленты учебных пособий.

Вычислительная техника

1. При изучении лекционного материала – лекционные залы с проекторами.
2. При проведении лабораторных занятий - работа на ПЭВМ.
3. При выполнении расчетно-графических работ – ПЭВМ.
4. При защите расчетно-графических работ – лекционные залы с проектора-ми.

Материальное обеспечение дисциплины

Лекционные залы, оборудованные проекторами, классы персональных ЭВМ. На-личие ЛВС и подключение к глобальной сети — обязательно.

Литература к лекциям и лабораторным работам

1. Техника Web-дизайна для студента [текст] / Ю.Е. Едомский. - СПб.:БХВ-Петербург, 2012. - 503 с. : рис., табл.
2. HTML и CSS. Самоучитель [текст] / Е. В. Мальчук. - СПб.:БХВ-Петер-бург, 2012. - 260 с. : рис.
3. Ваш первый сайт с использованием PHP-скриптов [текст] / Строганов А.С. - СПб.:БХВ-Петербург, 2013. - 411 с. : рис.
4. JavaScript. Энциклопедия пользователя [текст] / Ричард Вагнер, Аллен Вайк - СПб.:БХВ-Петербург, 2013. - 472 с. : рис.
5. Red Hat Linux для системных администраторов. Энциклопедия пользова-теля [текст] / Шенк Т. - К: Издательство «ДиаСофт», 2001, - 670 с. : рис.
6. Интернет-ресурсы: <http://www.opennet.org>, <http://www.javascript.ru>, [www.phpclub.ru](http://www.phpclub.ru), [www.w3c.org](http://www.w3c.org), <http://apache.org>

#### 4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

**Контроль и оценка** результатов освоения учебной дисциплины осуществляется преподавателем в процессе проведения практических занятий и лабораторных работ, тестирования, а также выполнения обучающимися индивидуальных заданий.

| Результаты обучения<br>(освоенные умения, усвоенные знания)   | Формы и методы контроля и оценки результатов обучения  |
|---|--|
| 1   | 2  |
| <p>уметь:</p> <ul style="list-style-type: none"> <li>– Проектировать информационные системы на языке C++</li> <li>– Проектировать и создавать программные системы Java</li> </ul> <p>знать</p> <ul style="list-style-type: none"> <li>– основ технологии объектно-ориентированной декомпозиции программных систем, базовых шаблонов проектирования (Наблюдатель, Итератор, Одиночка, Фабрика, Заместитель), отношений между классами и основ UML (диаграммы классов и последовательностей).</li> <li>– особенности построения объектно-ориентированных программных систем на C++.</li> <li>– Основные инструментальные средства языка C++ и стандартной библиотеки</li> <li>– Базовые знания платформы Java, особенности построения программных систем Java</li> <li>– Средства реализации принципов ООП и инструментальные средства языка Java.</li> <li>– Основы технологий построения простейших распределенных информационных систем и обеспечения безопасности.</li> </ul> | <p>экспертная оценка практической работы, обсуждение результатов выполнения самостоятельной работы, экспертная оценка контрольной работы</p> |

## МОДУЛЯ

Программа рассмотрена на заседании МЦК  
Экономики и права  
(протокол № от «\_\_» \_\_\_\_\_ 2024 г.)

Председатель МЦК  
Экономики и права  
«\_\_» \_\_\_\_\_ 2024 г. \_\_\_\_\_

Программа согласована на заседании МЦК  
Естественнонаучных дисциплин и  
информационных технологий  
(протокол № от «\_\_» \_\_\_\_\_ 2024 г.)

Председатель МЦК  
Естественнонаучных дисциплин и  
информационных технологий  
«\_\_» \_\_\_\_\_ 2024 г. \_\_\_\_\_