|  |  |  |  |
| --- | --- | --- | --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  РОССИЙСКОЙ ФЕДЕРАЦИИ  ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  «Национальный исследовательский ядерный университет «МИФИ»  Озерский технологический институт -  филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ»  (ОТИ НИЯУ МИФИ) | | | |
| Кафедра прикладной математики | | |
| УТВЕРЖДАЮ  ДИРЕКТОР  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И. А. Иванов  « 24 » мая 2021 г. | | |
| РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ | | |
| Объектно-ориентированное программирование | | |
| (наименование дисциплины ) | | |
|  | | |
| Направление подготовки (специальность): | 09.03.01 Информатика и вычислительная техника | |
|  |  | |
| Профиль подготовки: | Программное обеспечение средств вычислительной техники и автоматизированных систем | |
|  |  | |
| Наименование образовательной программы: | Программное обеспечение средств вычислительной техники и автоматизированных систем | |
|  |  | |
| Квалификация (степень) выпускника: | бакалавр |  |
| (бакалавр, магистр, специалист) |  |
|  |  | |
| Форма обучения: | очная |  |
| (очная, очно-заочная (вечерняя), заочная) |  |

г. Озёрск, 2021г.

# ЦЕЛИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Целью освоения учебной дисциплины «Объектно-ориентированное программирование» является изучение:

- базовых понятий ООП: абстракция, тип, класс, объект, свойство, метод, инкапсуляция;

- перегрузки функций и операторов;

- инициализации и уничтожения объектов;

- простого и множественного наследования;

- механизма виртуальных функций и полиморфизма;

- абстрактных классов и интерфейсов;

- механизма структурированных исключений;

- механизма шаблонов;

- особенностей языков Java и C#;

- механизма пакетов языка Java;

- делегирования событий в языках Java;

- многопоточного программирования в языках Java и C#;

- механизма делегатов и событий языка C#;

- изучение основ проектирования с помощью языка моделирования UML;

# МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВПО

Дисциплина «Объектно-ориентированное программирование» входит в Блок 1 «Дисциплины (модули)» в обязательную часть основной образовательной программы бакалавриата «Программное обеспечение средств вычислительной техники и автоматизированных систем», модуль «Профессиональный».

# КОМПЕТЕНЦИИ СТУДЕНТА, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ / ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ОБРАЗОВАНИЯ И КОМПЕТЕНЦИИ СТУДЕНТА ПО ЗАВЕРШЕНИИ ОСВОЕНИЯ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

В результате освоения данной дисциплины у студента формируются следующие компетенции и планируются следующие результаты обучения по дисциплине:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Компетенция / Индикатор** | | **Содержание** | **Результаты обучения по дисциплине** |
|  | | **ОПК-1** | **Способен использовать положения, законы и методы естественных наук и математики для решения задач инженерной деятельности** |  |
|  | | ОПК-1.1 | З-ОПК-1 Знать: основы математики, физики, общеинженерных знаний, вычислительной техники и программирования |  |
|  | | ОПК-1.2 | У-ОПК-1 уметь: решать нестандартные профессиональные задачи с применением естественнонаучных, математических и общеинженерных знаний |  |
|  | | ОПК-1.3 | В-ОПК-1 владеть: навыками решения нестандартных задач профессиональной деятельности, в том числе в междисциплинарном контексте |  |
|  | | **ОПК-8** | **Способен разрабатывать алгоритмы и программы, пригодные для практического применения** |  |  |
|  | | ОПК-8.1 | З-ОПК-8 Знать: алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения |  |  |
|  | | ОПК-8.2 | У-ОПК-8 Уметь: составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули |  |  |
|  | | ОПК-8.3 | В-ОПК-8 Владеть: языком программирования; навыками отладки и тестирования работоспособности программы |  |  |
|  | | **ПК-3** | **Способен разрабатывать модели и компоненты аппаратно-программных комплексов и баз данных, используя современные инструментальные средства и технологии** |  |  |
|  | | ПК-3.1 | З-ПК-3 Знать: схемотехнику логических схем, цифровых и запоминающих устройств, принципы построения и элементы микропроцессоров и микроконтроллеров, принципы работы программируемых логических матриц и программируемой матричной логики, основы объектно- ориентированного подхода к программированию, базы данных и системы управления базами данных для информационных систем различного назначения, принципы построения современных операционных систем и особенности их применения |  |  |
|  | | ПК-3.2 | У-ПК-3 Уметь: строить логические схемы счетчиков, регистров, сумматоров и запоминающих устройств, строить временные диаграммы работы интерфейсов и контроллеров, сопрягать аппаратные и программные средства в составе аппаратно- программных комплексов, работать с современными системами программирования, включая объектно- ориентированные |  |  |
|  | | ПК-3.3 | В-ПК-3 Владеть: современными инструментальными средствами проектирования цифровых устройств, языками процедурного и объектно- ориентированного программирования, навыками разработки и отладки программ |  |  |
|  | | **ПК-5.3** | **способен разрабатывать требования и проектировать программное обеспечение** |  |  |
|  | | ПК-5.3.1 | З-ПК-5.3 знать принципы, технологии, подходы и инструменты, используемые при разработке математического, информационного, технического, лингвистического, программного, эргономического, организационного и правового обеспечения программного обеспечения средств вычислительной техники и автоматизированных систем |  |  |
|  | | ПК-5.3.2 | У-ПК-5.3 уметь разрабатывать требования к математическому, информационному, техническому, лингвистическому, программному, эргономическому, организационному и правовому обеспечению программного обеспечения средств вычислительной техники и автоматизированных систем, проектировать его |  |  |
|  | | ПК-5.3.3 | В-ПК-5.3 владеть навыками разработки требований к математическому, информационному, техническому, лингвистическому, программному, эргономическому, организационному и правовому обеспечению программного обеспечения средств вычислительной техники и автоматизированных систем, навыками следования этим требованиям |  |  |

В результате изучения дисциплины студент должен ЗНАТЬ:

- синтаксис и типы данных языков программирования C++, Java и C#;

- основные понятия ООП: тип, класс, объект, свойство, метод, инкапсуляция;

­- назначение и применение виртуальных функций, абстрактных классов, шаблонов;

- модели делегирования событий в языках C# и Java;

- методы и средства многопоточного программирования в языках C# и Java;

В результате изучения дисциплины студент должен УМЕТЬ:

- моделировать объектно-ориентированные программные средства с помощью UML;

- анализировать предметную область и выявлять абстрактные типы данных;

- проектировать типы и иерархии типов в языках C++, C#, Java;

- проектировать шаблоны классов;

В результате изучения дисциплины студент должен ВЛАДЕТЬ:

- методами проектирования ООП систем на языке UML;

- методами делегирования событий в языках C# и Java;

- методами синхронизации потоков в языках C# и Java;

- методами структурированной обработки исключений;

# СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Общая трудоемкость дисциплины составляет | 8 | кредитов, |  |  |
| часов | 288 |  |  |  |

в том числе: контактная работа 172, самостоятельная работа 80, контроль 36

По семестрам

4 семестр: контактная работа 68 (лекции 34, практики 34), самостоятельная работа 40, зачет

5 семестр: контактная работа 68 (лекции 34, практики 34), КРП 36, самостоятельная работа 40, контроль 36 (экзамен)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| № п/п | Раздел учебной дисциплины | Недели | Виды учебной деятельности, включая СРС, трудоемкость (в часах) | | | | Текущий контроль успеваемости (*неделя, форма*) | Аттестация раздела  (*неделя, форма*) | Макс. балл за раздел |
|  |  |  | Лекции | Практ. занятия/ семинары | Лаб. раб. | СРС |  |  |  |
|  | 4 семестр |  |  |  |  |  |  |  |  |
| 1 | Основы объектно-ориентированной парадигмы | 1-2 | 4 | 4 | 0 | 6 | 1-2ПР | 5КР1  7ДЗ1 | 6 |
| 2 | Классы и объекты в C++ | 3-5 | 6 | 6 | 0 | 9 | 3-5ПР | 5КР1  7ДЗ1 | 9 |
| 3 | Специальные функции классов | 6-8 | 6 | 6 | 0 | 9 | 6-8ПР | 15ДЗ2 | 9 |
| 4 | Особенности классов C++ | 9-10 | 4 | 4 | 0 | 8 | 9-10ПР | 13КР2 | 8 |
| 5 | Наследование и полиморфизм | 11-14 | 8 | 8 | 0 | 10 | 11-14ПР | 15ДЗ2  КП | 12 |
| 6 | Структурированная обработка исключений | 15 | 2 | 2 | 0 | 1 | 15ПР | 15ДЗ2 | 3 |
| 7 | Потоковая модель ввода-вывода | 16 | 2 | 2 | 0 | 1 | 16ПР |  | 3 |
|  | Всего часов: | 1-16 | 32 | 32 | 0 | 44 | 108 |  |  |
|  | Итого баллов за семестр: |  |  |  |  |  |  |  | 50 |
|  | Зачет: |  |  |  |  |  |  |  | 50 |
|  | Итого за 4 семестр: |  |  |  |  |  |  |  | 100 |
|  | 5 семестр |  |  |  |  |  |  |  |  |
| 1 | Основные сведения о Java | 1-2 | 4 | 4 | 0 | 4 | 1-2ПР | 5КР3 | 5 |
| 2 | Классы, пакеты, интерфейсы, коллекции Java | 3-4 | 4 | 4 | 0 | 5 | 3-4ПР | 5КР3  8ДЗ3 | 5 |
| 3 | Модель делегирования событий в Java | 5-7 | 6 | 6 | 0 | 4,5 | 5-7ПР | 8ДЗ3 | 8 |
| 4 | Потоки в Java | 8-9 | 4 | 4 | 0 | 4 | 8-9ПР | 8ДЗ3 | 6 |
| 5 | Архитектура .NET Framework | 10-11 | 4 | 4 | 0 | 5 | 10-11ПР | 12КР4  16ДЗ4 | 6 |
| 6 | Классы C# | 12-14 | 6 | 6 | 0 | 5,5 | 12-14ПР | 12КР4  16ДЗ4 | 8 |
| 7 | Модель делегирования событий в C# | 15-16 | 4 | 4 | 0 | 4 | 15-16ПР | 16ДЗ4 | 6 |
| 8 | Потоки в C# | 17-18 | 4 | 4 | 0 | 4 | 17-18ПР | 16ДЗ4 | 6 |
|  | Всего часов: | 1-18 | 36 | 36 | 0 | 36 | 108 |  |  |
|  | Итого баллов за семестр: |  |  |  |  |  |  |  | 50 |
|  | Экзамен: |  |  |  |  |  |  |  | 50 |
|  | Итого за 5 семестр: |  |  |  |  |  |  |  | 100 |

Обозначения оценочных средств: ПР - практическая работа, КР - контрольная работа, ДЗ - индивидуальное домашнее задание, КП - курсовое проектирование.

Содержание разделов учебной дисциплины

4 семестр

|  |  |  |
| --- | --- | --- |
| № | Раздел учебной дисциплины | Содержание раздела |
| 1 | Основы объектно-ориентированной парадигмы | Парадигмы программирования. Абстракция в языках высокого уровня. Предпосылки возникновения ООП. Сущность ООП. Элементы объектной модели. Основные понятия ООП. ООП системы и языки. |
| 2 | Классы и объекты в C++ | Класс как абстракция. Класс как тип. Класс как структура данных. Класс как область видимости. Указатель this. Объекты. |
| 3 | Специальные функции классов | Конструктор класса. Деструктор класса. Сборка мусора. Операция присваивания. Конструктор копии. |
| 4 | Особенности классов C++ | Перегрузка операций. Друзья классов. Статические и константные элементы класса. |
| 5 | Наследование и полиморфизм | Наследование классов. Виртуальные функции. Полиморфизм классов и объектов. Таблица виртуальных методов. Механизм позднего связывания. Абстрактные классы. Интерфейсы. Шаблоны классов. |
| 6 | Структурированная обработка исключений | ООП подход к обработке исключений. |
| 7 | Потоковая модель ввода-вывода. | Потоковая модель ввода-вывода. |

5 семестр

|  |  |  |
| --- | --- | --- |
| № | Раздел учебной дисциплины | Содержание раздела |
| 1 | Основные сведения о Java | Исторические сведения о языке Java. Типы приложений. Компиляция и интерпретация программ Java. Байт-код. Основы языка Java. Типы данных, операторы и операции, массивы, перечисления и константы. |
| 2 | Классы, пакеты, интерфейсы, коллекции Java | Классы Java. Стандартные классы Math, String, Arrays. Оболочки простых типов. Пакеты Java. Интерфейсы Java. Коллекции Java. Обработка исключений в Java. |
| 3 | Модель делегирования событий в Java | Обработка событий. Классы событий. Классы-адаптеры событий. Анонимные классы обработчиков событий. Окна и элементы управления пакета AWT. Меню Java. Менеджеры компоновки. Диалоговые окна в Java. |
| 4 | Потоки в Java | Потоки и синхронизация потоков в Java. Ввод и вывод в Java. Байтовые потоки. Символьные потоки. Сериализация. Вспомогательные классы Java: лексический анализатор, битовые массивы, наблюдение за объектами. |
| 5 | Архитектура .NET Framework | История появления технологии .NET. Достоинства .NET. Архитектура .NET Framework, CLR, CTS, CLS. Компиляция программ .NET. Сборки, потоки и домены приложений .NET. Частные сборки и сборки общего доступа. Файлы конфигурации приложения. Пространства имен. |
| 6 | Классы C# | Язык C#. Типы данных. Модификаторы доступа к элементам классов. Операторы и операции. Ключевые слова аргументов, операций, классов и контекстов. Свойства и индексаторы классов. Обобщенные классы. |
| 7 | Модель делегирования событий в C# | Язык C#. Делегаты. События. |
| 8 | Потоки в C# | Использование потоков. Синхронизация потоков с помощью мониторов. Технология Windows Presentation Foundation. |

# ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Образовательный процесс по дисциплине строится на основе комбинации образовательных технологий. При освоении разделов дисциплины используется сочетание видов учебной деятельности (лекции, практические занятия, самостоятельная работа студента) с использованием интерактивных форм проведения занятий в аудитории.

Используемые образовательные технологии при изучении данной дисциплины:

- контекстное обучение;

- метод проектов;

- работа в команде;

- дискуссия;

- тренинг;

Интерактивные формы проведения занятий составляют 28 часов или 21% от общего объема аудиторных занятий.

# ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ.

## Текущий контроль проводится в виде контроля выполнения практических работ.

Примерные темы практических работ

4 семестр

1. Начальное конструирование классов.
2. Изучение простого наследования.
3. Иерархия классов.
4. Исследование иерархии классов. Конструкторы и деструкторы.
5. Виртуальные функции и полиморфизм.
6. Изучение отношения реализации между классами.
7. Изучение отношения ассоциации между классами.
8. Разработка класса стека.
9. Разработка класса дека.
10. Разработка класса очереди.
11. Объектно-ориентированный подход к обработке исключений.
12. Стандартные шаблоны Си++. Шаблон «стек».
13. Стандартные шаблоны Си++. Шаблон «дек».
14. Разработка иерархии типов: мой умный дом.

5 семестр

1. Изучение основ программирования на Java.
2. Изучение пакета AWT. События. Прослушивание событий.
3. Элементы управления Java.
4. Управление потоками и синхронизация потоков в Java.
5. Работа с изображениями в Java.
6. Работа с коллекциями в Java.
7. Работа с файлами в Java.
8. Программирование консольных приложений в C#.
9. Изучение структур C#.
10. Изучение классов C#.
11. Обработка исключений в C#.
12. Реализация системных интерфейсов C#.
13. Программирование индексаторов C#.
14. Изучение делегатов C#.
15. Программирование классов с событиями.
16. Изучение обобщенных классов C#.
17. Изучение управления сборками и приложениями.
18. Изучение программирования приложений с потоками.

## Рубежный контроль (аттестация раздела) проводится в виде контрольных работ и индивидуальных домашних заданий.

### Контрольная работа № 1, семестр 4.

Тема: Конструкторы и деструктор.

Время проведения - 5 неделя.

Варианты:

1. Конструктор по умолчанию.
2. Конструктор преобразования.
3. Конструктор копии.
4. Деструктор.

### Контрольная работа № 2, семестр 4.

Тема: Виртуальные функции и полиморфизм.

Время проведения - 13 неделя.

Варианты:

1. Виртуальные функции.
2. Таблица виртуальных методов.
3. Полиморфизм.
4. Абстрактные классы и интерфейсы.

### Контрольная работа № 3, семестр 5.

Тема: Разработка класса Java.

Время проведения - 5 неделя.

Варианты:

1. Класс «Авторучка».
2. Класс «Моторчик».
3. Класс «Фонарик».
4. Класс «Велосипед».
5. Класс «Пылесос».
6. Класс «Телевизор».
7. Класс «Домофон».
8. Класс «Кофеварка».

### Контрольная работа № 4, семестр 5.

Тема: Архитектура .NET Framework.

Время проведения - 12 неделя.

Варианты:

1. Общая среда выполнения.
2. Стандартная система типов.
3. Стандартная спецификация языка.
4. Компиляция программ .NET.

### Индивидуальное домашнее задание № 1, семестр 4.

Тема: Перегрузка операций.

Задание выдается на 3 неделе. Срок сдачи задания - 7 неделя.

Варианты:

1. Класс rat «Рациональное число».
2. Класс pos «Положительное число».
3. Класс nat «Натуральное число».
4. Класс fib «Число Фибоначчи».
5. Класс prime «Простое число».
6. Класс complex «Комплексное число».

### Индивидуальное домашнее задание № 2, семестр 4.

Тема: Класс использованием динамической памяти.

Задание выдается на 11 неделе. Срок сдачи задания - 15 неделя.

Варианты:

1. Класс string «Строка».
2. Класс darray «Динамический массив».
3. Класс dset «Динамическое множество».
4. Класс dlist «Динамический список».
5. Класс dstack «Динамический стек».
6. Класс ddeque «Динамический дек».
7. Класс dqueue «Динамическая очередь».

### Индивидуальное домашнее задание № 3, семестр 5.

Тема: Разработка простой игры на Java.

Задание выдается на 4 неделе. Срок сдачи задания - 8 неделя.

Варианты:

1. Игра «Змейка».
2. Игра «Теннис».
3. Игра «Судоку».
4. Игра «Шашки».
5. Игра «Крестики-нолики».
6. Игра «Морской бой».
7. Игра «Королевский квадрат».

### Индивидуальное домашнее задание № 4, семестр 5.

Тема: Разработка иерархии типов с использованием делегатов.

Задание выдается на 12 неделе. Срок сдачи задания - 16 неделя.

Варианты:

1. Иерархия типов «Магазин».
2. Иерархия типов «Автопарк».
3. Иерархия типов «Библиотека».
4. Иерархия типов «Фотоателье».
5. Иерархия типов «Почта».
6. Иерархия типов «Гостиница».
7. Иерархия типов «Поликлиника».
8. Иерархия типов «Кинотеатр».

## Промежуточная аттестация выполняется в виде зачета в 4 семестре, экзамена в 5 семестре и курсовой работы в 5 семестре.

### Примерный перечень вопросов к зачету

1. Отношения классов (обобщение, ассоциация, зависимость, реализация).
2. Инкапсуляция и целостность класса. Инкапсуляция и скрытие реализации.
3. Понятие класса. Класс как тип и как структура. Класс и представитель класса.
4. Виртуальные функции и полиморфизм. Таблица виртуальных методов.
5. Чистая виртуальная функция. Абстрактный класс. Понятие интерфейса.
6. Интерфейс и реализация класса. Раздельная компиляция.
7. Закрытая и открытая секции класса. Классы struct и class.
8. Конструктор класса. Деструктор класса. Перегрузка конструкторов в языке C++.
9. Конструктор копии. Моменты использования конструктора копии.
10. Конструкторы преобразования.
11. Операция присваивания.
12. Операции приведения к типу.
13. Статические элементы класса.
14. Друзья класса: друзья-функции и друзья-классы.
15. Шаблон функции. Шаблон класса.
16. Простое наследование C++. Ключевое слово protected.
17. Множественное наследование. Неоднозначность при множественном наследовании.
18. Управление исключениями в C++.
19. Объектно-ориентированный подход к управлению исключениями.

### Примерный перечень вопросов к экзамену

1. Модели программирования (физическая, процедурная, структурная).
2. Класс как абстракция.
3. Класс как структура данных.
4. Класс как тип.
5. Класс как область действия.
6. Класс и экземпляр класса.
7. Интерфейс и реализация класса. Файлы и модули класса.
8. Инкапсуляция элементов класса. Инкапсуляция и целостность типа.
9. Элементы данных, функции-элементы, свойства и методы класса.
10. Наследование в Си++. Неоднозначность при множественном наследовании.
11. Виртуальная функция. Таблица виртуальных методов.
12. Механизм позднего связывания. Полиморфизм.
13. Чистая виртуальная функция. Абстрактный класс.
14. Статические элементы класса.
15. Шаблоны классов.
16. Друзья классов.
17. Объектно-ориентированный подход к управлению исключениями.
18. Конструктор по умолчанию. Конструктор преобразования.
19. Конструктор копии.
20. Деструктор класса.
21. Язык Java. Типы программ.
22. Язык Java. Типы данных.
23. Язык Java. Классы.
24. Язык Java. Оболочки простых типов.
25. Язык Java. Коллекции.
26. Язык Java. Пакеты.
27. Язык Java. Обработка событий.
28. Язык Java. Менеджеры компоновки.
29. Язык Java. Управление потоками.
30. Язык Java. Синхронизация потоков.
31. DOT NET. Архитектура. CLR, CTS, CLS.
32. DOT NET. Компиляция программ.
33. DOT NET. Сборки частного доступа. Файлы конфигурации.
34. DOT NET. Сборки общего доступа. GAC.
35. Язык C#. Типы данных.
36. Язык C#. Модификаторы.
37. Язык C#. Свойства.
38. Язык C#. Индексаторы.
39. Язык C#. Делегаты.
40. Язык C#. События.

### Курсовое проектирование

Примерные темы курсовых работ:

1. Иерархия типов: минералы.
2. Иерархия типов: строительные материалы.
3. Иерархия типов: строительный инструмент.
4. Иерархия типов: музыкальные инструменты.
5. Иерархия типов: бабочки.
6. Иерархия типов: насекомые.
7. Иерархия типов: травоядные.
8. Иерархия типов: млекопитающие.
9. Иерархия типов: летающие аппараты.
10. Иерархия типов: напитки.
11. Иерархия типов: лекарственные растения.
12. Иерархия типов: машиностроительные механизмы.
13. Иерархия типов: сортамент (металлопрокат).
14. Иерархия типов: электронные приборы.
15. Иерархия типов: измерительные приборы.
16. Иерархия типов: метрология.

## Самостоятельная работа студента

|  |  |  |
| --- | --- | --- |
| Раздел учебной дисциплины | Виды СРС | Часов |
| 4 семестр |  |  |
| Все | ПЛ1-16, ПП1-16 | 16 |
| Основы объектно-ориентированной парадигмы | ПК1, ДЗ1 | 4 |
| Классы и объекты в C++ | ПК1, ДЗ1 | 6 |
| Специальные функции классов | ПК2, ДЗ2 | 6 |
| Особенности классов C++ | ПК2, ДЗ2 | 6 |
| Наследование и полиморфизм | ПК2, ДЗ2 | 6 |
|  | Всего часов: | 44 |
| 5 семестр |  |  |
| Все | ПЛ1-18, ПП1-18 | 9 |
| Все | КП | 16 |
| Основные сведения о Java | ПК3 | 1 |
| Классы, пакеты, интерфейсы, коллекции Java | ПК3, ДЗ3 | 2 |
| Модель делегирования событий в Java | ДЗ3 | 1 |
| Потоки в Java | ДЗ3 | 1 |
| Архитектура .NET Framework | ПК4, ДЗ4 | 2 |
| Классы C# | ПК4, ДЗ4 | 2 |
| Модель делегирования событий в C# | ДЗ4 | 1 |
| Потоки в C# | ДЗ4 | 1 |
|  | Всего часов: | 36 |

ДЗ — индивидуальное домашнее задание, ПЛ — подготовка к лекциям, ПП — подготовка к практическим занятиям, ПК — подготовка к контрольной работе, КП - курсовое проектирование.

# УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

## Основная литература:

Пышкин Е.В. Основные концепции и механизмы объектно-ориентирован­ного программирования. - СПб.: БХВ-Петербург, 2005. - 640 с.: ил.

Васильев А.Н. Объектно-ориентированное программирование: Учебное пособие. - СПб: Питер: 2014. - 400 с. - (Серия «Учебное пособие»).

Пономарев В. В. Система программирования .NET и язык C#. Учебно-методическое пособие. Озёрск: ОТИ НИЯУ МИФИ, 2014. - 40 с., ил.

Пономарев В.В. Объектно-ориентированное программирование на C++. Учебно-методи­ческое пособие. Озёрск: ОТИ НИЯУ МИФИ, 2014. - 60 с., ил.

## Дополнительная литература:

Пономарев В. В. Введение в Java для программистов. Учебно-методическое пособие. Озёрск: ОТИ НИЯУ МИФИ, 2013. - 58 с., ил.

Пономарев В.В. ООП на C++: Метод. пособие. - Озерск: ОТИ МИФИ, 2005. - 71 с.

Подбельский В. В. Язык Си++: Учеб. пособие. - 3-е изд., дораб. - М.: Финансы и статистика, 1998. - 560 с.: ил.

Синяков В.Е. Объектно-ориентированный анализ и проектирование: Методические указания / В.Е. Синяков. - Озерск: ОТИ МИФИ, 2007. - 40 с.

Лафоре Р. Объектно-ориентированное программирование в С++/ Р. Лафоре. - 4-е изд. - СПб; М; Нижний Новгород [и др.]: Питер, 2013. - 923 с. - (Классика Computer Science).

## Программное обеспечение и Интернет-ресурсы:

http://www.intuit.ru/studies/courses/71/71/info (Национальный открытый университет «ИНТУИТ», курс «Основы объектно-ориентированного программирования»).

http://www.intuit.ru/studies/courses/72/72/info (Национальный открытый университет «ИНТУИТ», курс «Основы объектно-ориентированного проектирования»).

# МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

## Лекции проводятся в аудиторном классе, оборудованном доской.

## Практические занятия проводятся в компьютерном классе (11 компьютеров).

Требуемое программное обеспечение

|  |  |  |
| --- | --- | --- |
| № | Программный продукт | Количество |
| 1 | Операционная система Microsoft Windows XP, 7, 8 | 1 шт. на компьютер |
| 2 | Среда программирования Microsoft Visual Studio .NET | 1 шт. на компьютер |
| 3 | Файловый менеджер FAR | 1 шт. на компьютер |
| 4 | Пакет Java SDK | 1 шт. на компьютер |
| 5 | Среда разработки IntelliJIdea 14 | 1 шт. на компьютер |

Программа составлена в соответствии с требованиями ОС НИЯУ МИФИ по направлению подготовки (специальности):

|  |  |
| --- | --- |
|  | |
| Автор | А. Ф. Зубаиров |
| Рецензент | С.А.Балакин,  инженер-программист СИТ ФГУП «ПО «МАЯК» |
| Программа одобрена на заседании  методического совета кафедры | 24.05.2021 протокол №5 |