

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

**Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Институт информационных технологий, математики и механики

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол от
30.11.2022г. №13

Рабочая программа дисциплины (модуля)

**Объектноориентированное
программирование**

Уровень высшего образования
бакалавриат

Направление подготовки / специальность
09.03.03 Прикладная информатика

Направленность образовательной программы
Прикладная информатика в информационной сфере

Форма обучения
очная

Нижний Новгород

2022

1. Место и цели дисциплины (модуля) в структуре ОПОП

Дисциплина относится к части, формируемой участниками образовательных отношений.

№ варианта	Место дисциплины в учебном плане образовательной программы	Стандартный текст для автоматического заполнения в конструкторе РПД
2	Блок 1. Дисциплины (модули) Часть, формируемая участниками образовательных отношений	Дисциплина Б1.В.12 «Объектноориентированное программирование» относится к части ООП направления подготовки 09.03.03 Прикладная информатика, формируемой участниками образовательных отношений.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции* (код, содержание индикатора)	Результаты обучения по дисциплине**	
ПК-8. Способен разрабатывать лингвистическое, информационное и программное обеспечение ИС (ИИС) и сопровождающую его документацию	ПК-8.1. Демонстрирует знание современных языков и систем программирования, формализмов описания знаний на концептуальном и инфологическом уровнях, требований к технической документации на все виды обеспечения ИС (ИИС).	Знать: парадигмы объектноориентированного программирования специальные методы классов (конструкторы и их типы и деструктор)	Собеседование Тест
	ПК-8.2. Применяет современные языки и системы программирования, формализмы описания знаний на концептуальном и инфологическом уровнях при разработке лингвистического, информационного и программного обеспечения ИИС и сопровождающей ее документации.	Уметь формировать объявления и описания классов (на примере C++) использовать при объявлении и описании классов все типы конструкторов и деструктор класса (на примере C++) применять механизмы возбуждения и обработки исключений	лабораторная работа задача
	ПК-8.3. Имеет практический опыт разработки лингвистического, информационного и программного обеспечения конкретной ИС (ИИС) и сопровождающей ее документации.	Владеть основными элементами технологии применения парадигм объектноориентированного программирования	лабораторная работа
ПК-9. Способен моделировать	ПК-9.1. Демонстрирует знание методических основ	Знать особенности механизма перегрузки	Собеседование Тест

прикладные (бизнес) процессы и объекты предметной области	моделирования процессов и объектов предметной области.	операций особенности механизма наследования классов и перекрытия методов при наследовании принципы объявления и описания шаблонов функций и шаблонов классов	
	ПК-9.2. Демонстрирует умение применения знаний к моделированию прикладных процессов и объектов предметной области при разработке программного обеспечения ИС.	Уметь строить схемы наследования с перекрытием методов классов использовать механизмы виртуального перекрытия методов применять шаблоны функций и шаблоны классов	лабораторная работа
	ПК-9.3. Имеет практический опыт моделирования процессов и объектов на примере конкретной предметной области.	Владеть основами проектирования классов и организации взаимодействия объектов методами перегрузки отдельных типов операций	лабораторная работа

3. Структура и содержание дисциплины

3.1. Трудоемкость дисциплины

	очная форма обучения
Общая трудоемкость	8 ЗЕТ
Часов по учебному плану	288
в том числе	
контактная работа:	115
- занятия лекционного типа	48
- занятия лабораторного типа	64
- текущий контроль (КСР)	3
самостоятельная работа	137
Промежуточная аттестация –зачет, экзамен	36

3.2. Содержание дисциплины

Наименование и краткое содержание разделов и тем дисциплины, форма промежуточной аттестации по дисциплине	Всего (часы)	в том числе				Самостоятельная работа студента часы
		контактная работа (работа во взаимодействии с преподавателем), часы				
		Занятия лекционного типа	Занятия семинарского типа	Лабораторные работы	Всего контактных часов	
Отличия языка C++ от языка C	15	2		4	6	9
Введение в объектно-ориентированное	16	2		4	6	10

программирование						
Классы и объекты	18	2		6	8	10
Конструкторы и деструктор	20	4		6	10	10
Обработка исключений	18	2		6	8	10
Перегрузка операций	20	4		6	10	10
Текущий контроль (КСР)	1				1	
Промежуточная аттестация (зачет)						
Агрегация и наследование	36	8		8	16	20
Виртуальные методы	36	8		8	16	20
Шаблоны функций и классов	44	12		12	24	20
Битовые операции	26	4		4	8	18
Текущий контроль (КСР)	2				2	
Промежуточная аттестация (экзамен)	36					
Итого	288	48		64	115	137

Текущий контроль успеваемости реализуется в форме опросов на занятиях лабораторного типа.

Промежуточная аттестация проходит в традиционной форме (зачет и экзамен).

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа студента при изучении дисциплины включает :

1. Изучение учебной литературы и лекционных материалов.
2. Выполнение домашних практических заданий.
3. Удаленное тестирование Методы объектно-ориентированного программирования.
URL: <http://e-learning.unn.ru/course/view.php?id=251>
4. Подготовка к промежуточной аттестации в форме зачета.
5. Подготовка к промежуточной аттестации в форме экзамена.

Контрольные вопросы и задания для проведения текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведены в п. 5.2.

5. Фонд оценочных средств для промежуточной аттестации по дисциплине, включающий:

5.1. Описание шкал оценивания результатов обучения по дисциплине

Уровень сформированности компетенций (индикатора достижения)	Шкала оценивания сформированности компетенций					
	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	превосходно
	не зачтено		зачтено			

компетенций)							
<u>Знания</u>	Отсутствие знаний теоретического материала.	Уровень знаний ниже минимальных требований. Имели место грубые ошибки.	Минимально допустимый уровень знаний. Допущено много негрубых ошибки.	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок.	Уровень знаний в объеме, превышающем программу подготовки.
	Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа						
<u>Умения</u>	Отсутствие минимальных умений.	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки.	Продemonстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания но не в полном объеме.	Продemonстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме, но некоторые с недочетами.	Продemonстрированы все основные умения, решены все основные задачи с отдельными недочетами, выполнены все задания в полном объеме.	Продemonстрированы все основные умения,. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
	Невозможность оценить наличие умений вследствие отказа обучающегося от ответа						
<u>Навыки</u>	Отсутствие владения материалом.	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки.	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продemonстрированы базовые навыки при решении стандартных задач без ошибок и недочетов.	Продemonстрированы навыки при решении нестандартных задач без ошибок и недочетов.	Продemonстрированы творческий подход к решению нестандартных задач
	Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа						

Шкала оценки при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	Превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно»

	Отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично»
	Очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо»
	Хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо»
	Удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	Неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо»
	Плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения

5.2.1. Типовые вопросы к экзамену по дисциплине «Объектно ориентированное программирование»

Вопросы	Код формируемой компетенции
Абстрактные типы данных. Понятие класса. Инкапсуляция. Объявление класса. Поля и методы. Скрытие членов класса: секции <code>public</code> и <code>private</code> . Описание класса. Классы и структуры. Пример разработки класса «Комплексное число».	ПК-8
Объявление объектов класса. Указатели на объекты. Взаимодействие объектов и методов класса. Пример разработки класса «Строка».	ПК-8
Конструкторы. Типы конструкторов. Конструкторов классов «Комплексное число» и «Строка». Когда и как вызываются конструкторы.	ПК-8
Конструкторы копирования и преобразования типов. Примеры разработки конструкторов копирования и преобразования типов.	ПК-8
Деструкторы. Когда и как вызываются деструкторы.	ПК-8
Константные поля. Инициализация константных полей. Константные методы. Примеры константных полей и методов. Константные объекты.	ПК-8
Статические поля и методы. Примеры статических полей и методов.	ПК-8
Перегрузка операций. Общие правила и ограничения. Два способа перегрузки операций. Дружественные функции класса.	ПК-9
Правила перегрузки операции присваивания.	ПК-9
Правила перегрузки операций <code>+</code> , <code>-</code> , <code>*</code> , <code>/</code> .	ПК-9

Правила перегрузки операций типа +=.	ПК-9
Перегрузка операции индексации и унарных операций.	ПК-9
Перегрузка операции ввода/вывода в поток.	ПК-9
Перегрузка операций и конструкторы преобразования типов. Класс CData	ПК-9

5.2.2. Типовые вопросы к зачету по дисциплине «Объектноориентированное программирование»

Вопрос	Код компетенции (согласно РПД)
Парадигмы ООП. Примеры применения парадигм ООП.	ПК-8
Абстрактные типы данных. Понятие класса. Инкапсуляция. Объявление класса. Поля и методы. Скрытие членов класса: секции public и private. Описание класса. Классы и структуры. Пример разработки класса “Комплексное число”.	ПК-8
Объявление объектов класса. Указатели на объекты. Взаимодействие объектов и методов класса. Пример разработки класса “Строка”.	ПК-8
Перегрузка операций. Общие правила и ограничения. Два способа перегрузки операций. Дружественные функции класса.	ПК-9
Правила перегрузки операции присваивания.	ПК-9
Правила перегрузки операций +, -, *, /.	ПК-9
Правила перегрузки операций типа +=.	ПК-9

5.2.3. Типовые тестовые задания для оценивания результатов формирования компетенции ПК-8

1. \$БООП_1_1. Укажите признаки парадигмы «Инкапсуляция»:

данные объединяются с алгоритмами в объекты: члены-данные (поля) и члены-методы + сокрытие деталей - поля и методы объекта делятся на открытые (интерфейс объекта) и скрытые +

объекты могут наследоваться от других объектов

при наследовании потомок: сохраняет свойства и методы предка, добавляет новые свойства и методы, может менять старые

определение поведения объекта контекстом его использования
2. \$БООП_1_2. Укажите признаки парадигмы «Наследование»:

данные объединяются с алгоритмами в объекты: члены-данные (поля) и члены-методы сокрытие деталей - поля и методы объекта делятся на открытые (интерфейс объекта) и скрытые

объекты могут наследоваться от других объектов +

при наследовании потомок: сохраняет свойства и методы предка, добавляет новые свойства и методы, может менять старые +

определение поведения объекта контекстом его использования

3. \$BOOP_1_3. Укажите признаки парадигмы «Полиморфизм»:
данные объединяются с алгоритмами в объекты: члены-данные (поля) и члены-методы
сокрытие деталей - поля и методы объекта делятся на открытые (интерфейс объекта) и скрытые
объекты могут наследоваться от других объектов
при наследовании потомок: сохраняет свойства и методы предка, добавляет новые свойства и методы, может менять старые
определение поведения объекта контекстом его использования +
4. \$КиО_1_1. Класс это:
специальный тип данных +
структура специального вида
переменные специального вида
5. \$КиО_1_2. Объявление класса включает:
объявление имени класса +
объявления полей класса +
объявления методов класса +
объявления переменных класса
объявления констант

5.2.4. Типовые тестовые задания для оценивания результатов формирования компетенции ПК-9

1. \$ПерО_1_1. Укажите правильные варианты объявления функций перегрузки операции для класса T:
T operator+(const T& t); +
bool operator==(int p, const T& t); +
T operation*(const T& t, int p); +
T operation/(const T& t1, const T& t2); +
T operator*(T& t);
T operation=(const T& t);
bool operator<(T& t);
2. \$ПерО_1_2. Выберите правильные утверждения:
функция перегрузки операции может быть внешней (не методом класса) +
метод класса, перегружающий операцию может быть объявлен как virtual +
функция перегрузки операции должна быть методом класса
метод класса, перегружающий операцию может быть объявлен как static
3. \$ПерО_1_3. Выберите правильные утверждения:
внешняя функция, перегружающая бинарную операцию, должна иметь два параметра +
метод класса, перегружающий бинарную операцию, первый операнд получает как this +
метод класса, перегружающий бинарную операцию, должен иметь два параметра
4. \$ПерО_2_1. Какие из приведенных ниже операций не могут быть перегружены:
:: (область видимости) +
sizeof (размер типа, объекта) +
new (запрос памяти)
* (разыменовывание)
5. \$ПерО_2_2. Какие из приведенных ниже операций могут быть перегружены только как метод класса:
-> (косвенный выбор члена) +

() (вызов функции) +
new (запрос памяти)
* (разыменовывание)
++ (инкремент)

5.2.4. Типовые задания (лабораторные работы) для оценивания результатов формирования компетенции ПК-8

1. Создать класс Fraction для работы с дробными числами. Число должно быть представлено двумя полями: целая часть — длинное целое со знаком, дробная часть — беззнаковое короткое целое. Реализовать арифметические операции сложения, вычитания, умножения и операции сравнения.
2. Создать класс vector3D, представления вектора в 3-х мерном пространстве: x, y, z. Обязательно должны быть реализованы методы: сложение и вычитание векторов, скалярное произведение векторов, умножение на скаляр, сравнение векторов, вычисление длины вектора, сравнение длины векторов.
3. Создать класс Money для работы с денежными суммами. Число должно быть представлено двумя полями: типа long для рублей и типа unsigned char — для копеек. Дробная часть (копейки) при выводе на экран должна быть отделена от целой части запятой. Реализовать сложение, вычитание, деление суммы на дробное число, умножение на дробное число и операции сравнения.
4. Создать класс Point для работы с точками на плоскости. Координаты точки — декартовы. Обязательно должны быть реализованы: перемещение точки по оси X, перемещение по оси Y, определение расстояния до начала координат, расстояния между двумя точками, преобразование в полярные координаты, сравнение на совпадение и несовпадение.

5.2.5. Типовые лабораторные работы для оценивания результатов формирования компетенции ПК-9

1. Задача Liquid

1. Создать класс Liquid (жидкость), имеющий поля названия и плотности. Определить методы переназначения и изменения плотности. Определить необходимые конструкторы, вывод параметров объекта на консоль.
2. Создать производный класс Alcohol (спирт), имеющий крепость. Определить методы переназначения и изменения крепости. Определить необходимые конструкторы, вывод параметров объекта на консоль.
3. Из класса Alcohol создать производный класс Beer (пиво), имеющий процент содержания хмеля. Определить методы переназначения и изменения крепости и процента хмеля. Определить необходимые конструкторы, вывод параметров объекта на консоль.

2. Задача Man

1. Создать класс Man (человек), с полями: имя, возраст, пол и вес. Определить методы переназначения имени, изменения возраста и изменения веса. Определить необходимые конструкторы, вывод параметров объекта на консоль.
2. Создать производный класс Student, имеющий поля факультет, курс, группа. Определить методы изменения перехода на следующий курс, перевода в другую группу. Определить необходимые конструкторы, вывод параметров объекта на консоль.

3. Задача Pair/ Fraction

1. Создать класс Pair (пара целых чисел); определить методы изменения полей и сравнения пар: пара p1 больше пары p2, если (first.p1 > first.p2) или (first.p1 = first.p2) и (second.p1 > second.p2). Определить необходимые конструкторы, вывод параметров объекта на консоль.

2. Определить класс-наследник Fraction с полями: целая часть числа и дробная часть числа. Определить полный набор методов сравнения. Определить необходимые конструкторы, вывод параметров объекта на консоль

5.2.6. Типовые задачи, выносимые на экзамен по дисциплине «Объектноориентированное программирование (C++)»

1. Написать объявление класса Person. Полями класса должны быть:

- ФИО
- Адрес
- Возраст
- Телефон

Класс должен иметь все необходимые конструкторы и деструктор. Должны быть перегружены операции присваивания, сравнения и обмена с потоком.

Привести описание перегрузки операции присваивания.

2. Написать объявление класса Person. Полями класса должны быть:

- ФИО
- Адрес
- Возраст
- Телефон

Класс должен иметь все необходимые конструкторы и деструктор. Должны быть перегружены операции присваивания, сравнения и обмена с потоком.

Привести описание перегрузки операций обмена с потоком.

3. Написать объявление класса Person. Полями класса должны быть:

- ФИО
- Адрес
- Возраст
- Телефон

Класс должен иметь все необходимые конструкторы и деструктор. Должны быть перегружены операции присваивания, сравнения и обмена с потоком.

Привести описание конструктора копирования.

4. Написать объявление класса Time для работы с временем. Полями класса должны быть:

- часы
- минуты
- секунды

Класс должен иметь все необходимые конструкторы и деструктор. Должны быть перегружены операции присваивания, сравнения, увеличения (уменьшения) времени на указанное к-во секунд и обмена с потоком.

Привести описание операций присваивания, сравнения и увеличения времени.

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

а) Основная литература:

1. Карпенко С.Н. и др. Методы объектно-ориентированного программирования. URL: <http://e-learning.unn.ru/course/view.php?id=251>
2. Bertrand Meyer Основы объектно-ориентированного программирования. НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/71/71/info>
3. Ирина Белоцерковская, Наталья Галина, Лилия Катаева Алгоритмизация. Введение в язык программирования C++. НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/16740/1301/info>
4. Юлий Кетков Введение в языки программирования C и C++. НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/1039/231/info>

б) Дополнительная литература:

1. Сергей Зыков Введение в теорию программирования. Объектно-ориентированный подход НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/50/50/info>
2. Роберт Седжвик Алгоритмы на C++. НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/12181/1174/info>
3. Бьерн Страуструп. Язык программирования C++ для профессионалов. НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/98/98/info>
4. Татьяна Павловская Программирование на языке C++. НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/626/482/info>
5. Александр Фридман Язык программирования C++. НОУ ИНТУИТ. URL: <http://www.intuit.ru/studies/courses/17/17/info>

7. Материально-техническое обеспечение дисциплины (модуля)

Помещения представляют собой учебные аудитории для проведения учебных занятий, предусмотренных программой (лекционного типа), оснащенные оборудованием и техническими средствами обучения.

Помещения для самостоятельной и лабораторной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки 09.03.03 «Прикладная информатика».

Авторы: доцент _____ Карпенко С.Н. ,

доцент _____ Маркина М.В.

Рецензент профессор _____ Федосенко Ю.С.

Заведующий кафедрой _____ Золотых Н.Ю.

Программа одобрена на заседании методической комиссии института информационных технологий, математики и механики

07.12.2022 протокол №4