

государственное бюджетное образовательное учреждение  
высшего образования Московской области  
«Университет «Дубна»  
(государственный университет «Дубна»)

Институт системного анализа и управления  
Кафедра распределенных информационно-вычислительных систем



**Рабочая программа дисциплины**  
**Объектно-ориентированное программирование**

Направление подготовки (специальность)  
**01.03.02 Прикладная математика и информатика**

Уровень высшего образования  
**Бакалавриат**

Направленность (профиль) программы (специализация)  
**Математическое моделирование**

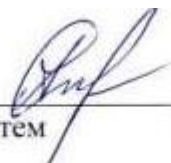
Форма обучения  
**очная**

Дубна, 2021

Преподаватель:

доктор физ.-мат. наук, профессор Смирнов А. В.

кафедра Распределенных информационно-вычислительных систем

/  /  
подпись

Рабочая программа разработана в соответствии с требованиями ФГОС ВО по направлению подготовки высшего образования **01.03.02 Прикладная математика и информатика**

Программа рассмотрена на заседании кафедры **распределенных информационных вычислительных систем**

Протокол заседания № 12 от «11» июня 2021 г.

Заведующий кафедрой д.т.н. профессор Кореньков В.В.

  
подпись

СОГЛАСОВАНО

Заведующий выпускающей кафедрой распределённых информационно-вычислительных систем Кореньков В.В.

  
подпись

Эксперт (рецензент):

Помощник директора лаборатории информационных технологий имени М.Г. Мещерякова Объединенного института ядерных исследований по международному сотрудничеству и работе с кадрами, к.ф.-м.н., с.н.с., Айрян Э.А.

  
подпись



### **Цели и задачи освоения дисциплины**

Цель дисциплины – приобретение студентами теоретических знаний в области идеологии ООП и устойчивых навыков практической работы в среде объектно-ориентированного программирования.

Задачами являются системное освоение основных понятий ООП и овладение технологическими средствами ООП.

Дать студентам **теоретические знания** основ ООП:

- базовых понятий и компонентов ООП, их взаимосвязей;
- основных правил и приемов разработки программного продукта с помощью технологии ООП;
- преимуществ ООП перед прежними технологиями создания программного обеспечения;
- достоинств и недостатков технологии ООП.

Выработать и развить **практические умения и навыки**:

- разработки собственных классов, ориентированных на решение поставленной задачи;
- проектирования и реализации иерархии классов.
- реализации соответствующей структуре спроектированных данных и алгоритмов их обработки;

Дисциплина должна обеспечить знакомство студента с базовыми положениями ООП, основными стандартными компонентами, методикой разработки программ в ООП, основными технологическими средствами ООП.

В результате освоения курса студент должен обладать навыками разработки и программной реализации алгоритмов на основе объектно-ориентированной технологии программирования.

## **2. Объекты профессиональной деятельности при изучении дисциплины**

Объектами профессиональной деятельности в рамках изучаемой дисциплины являются: информатика и управление; теория, алгоритмы, приложения; математическое и информационное обеспечение экономической деятельности; математические методы и программное обеспечение защиты информации; математическое и программное обеспечение компьютерных сетей; информационные системы и их исследование методами математического прогнозирования и системного анализа; интеллектуальные системы; программная инженерия; системы управления предприятием; сетевые технологии.

## **3. Место дисциплины в структуре ОПОП бакалавриата**

Учебный курс относится к разделу «Обязательные дисциплины вариативной части» учебного плана. Изучается в 3 семестре, форма промежуточной аттестации – зачет с оценкой.

Для успешного овладения курса Объектно-ориентированное программирование студенты должны иметь удовлетворительные знания, читаемые в учебных курсах

Программирование на языке высокого уровня, Компьютерный практикум, Математический анализ, Линейная алгебра и аналитическая геометрия.

**Формы работы студентов** в ходе изучения дисциплины предусмотрены лекционные, семинарские занятия, выполнение домашних работ.

**Самостоятельная работа студентов**, предусмотренная учебным планом, выполняется в ходе семестра в форме завершения проектов, теоретически проанализированных на семинарском занятии и выполнение этапов которых начато на семинарских занятиях.

Отдельные темы теоретического курса прорабатываются студентами самостоятельно в соответствии с планом самостоятельной работы и конкретными заданиями преподавателя с учетом индивидуальных особенностей студентов.

**Виды текущего контроля** – проверка домашних заданий, индивидуальная защита выполненных проектов.

**Форма итогового контроля: зачет с оценкой**

#### **4. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями выпускников)**

После изучения данной дисциплины студенты приобретают знания, умения и владения (навыки), соответствующие результатам основной профессиональной образовательной программы.

<b>Формируемые компетенции</b> <i>(код компетенции, уровень (этап) освоения)</i> <i>(последний – при наличии в карте компетенции)</i>	<b>Планируемые результаты обучения по дисциплине (модулю), характеризующие этапы формирования компетенций</b>
ОПК-3 – способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям	31(ОПК-3) Знать: основные парадигмы и языки программирования 33(ОПК-3) Знать: базовые структуры данных и основные алгоритмы их обработки 34(ОПК-3) Знать: основы создания компонент программного обеспечения У3(ОПК-3) Уметь: выбирать алгоритмы и выполнять их программную реализацию для решения типовых задач предметной области В1(ОПК-3) Владеть: современными интегрированными средами разработки программного обеспечения В2(ОПК-3) Владеть: навыками тестирования, отладки и верификации

	программ
ОПК-4 – способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	34(ОПК-4) Знать: Базовые знания математики, информатики, программирования и информационных технологий
ПК-2 – способностью понимать, совершенствовать и применять современный математический аппарат	В1(ПК-2) Владеть: кодированием на алгоритмических языках
ПК-17 – способностью к разработке и применению алгоритмических и программных решений, математических методов обработки данных в области профессиональной деятельности	<p>31(ПК-17) Знать: формальные методы и инструменты разработки программного продукта</p> <p>32(ПК-17) Знать: общие принципы разработки программных средств</p> <p>34(ПК-17) Знать: способы оптимизации программного кода**)</p> <p>36(ПК-17) Знать: методы построения и оценки эффективности алгоритмов и их программной реализации</p> <p>У6(ПК-17) Уметь: разрабатывать пользовательский интерфейс, прикладной программный интерфейс, в соответствии с постановкой задачи *)</p>

\*) Результат обучения сформулирован на основании требований профессионального стандарта: 06.001 «Программист», для выполнения обобщенной трудовой функции D: «Разработка требований и проектирование программного обеспечения»

\*\*) Результат обучения сформулирован на основании требований профессионального стандарта: 06.015 «Специалист по информационным системам» для выполнения обобщенной трудовой функции C: «Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы»

## **5. 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 составляет 3 зачетных единицы, всего 108 часов, из которых:

— 52 час составляет контактная работа обучающегося с преподавателем<sup>1</sup>:

<sup>1</sup> Перечень видов учебных занятий уточняется в соответствии с учебным планом.

- 18 часов – лекционные занятия,
- 34 часов – практические занятия,
- **56 часов составляет самостоятельная работа обучающегося.**
- мероприятия промежуточной аттестации — (зачет с оценкой в 3 семестре).

**6. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических или астрономических часов и виды учебных занятий**

7.

Наименование и краткое содержание разделов и тем дисциплины Форма промежуточной аттестации по дисциплине	Всего (часы )	Контактная работа (работа во взаимодействии с преподавателем), часы из них <sup>2</sup>					Самостоятельная работа обучающегося, часы, из них			
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*	Всего	Выполнение домашних заданий	Подготовка рефератов и т.п.	Всего
Объектный подход к разработке программного обеспечения. Объекты: абстракция, ограничение доступа, модульность, иерархия, типизирование, параллелизм, устойчивость. Идентификация классов и объектов. Примеры объектного моделирования реальных систем. Концепция типов данных. Стандартные типы данных. Типы данных, определяемые пользователем. Упаковка и распаковка переменных	16	2		6		ПР-1	8	8		8
Понятия класса и объекта. Общее и различие в понятиях класс и объект. Взаимоотношение класса и объекта. Члены класса. Поля и методы класса. Общие (public) и закрытые	6	1		3		ПР-1	4	2		2

<sup>2</sup> Перечень видов учебных занятий уточняется в соответствии с учебным планом.

Наименование и краткое содержание разделов и тем дисциплины Форма промежуточной аттестации по дисциплине	Всего (часы)	Контактная работа (работа во взаимодействии с преподавателем), часы из них <sup>2</sup>					Всего	Самостоятельная работа обучающегося, часы, из них		
		Лекционные занятия	Семинарские занятия	Практические занятия	Лабораторные занятия	Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*		Выполнение домашних заданий	Подготовка рефератов и т.п.	Всего
(private) поля и методы класса. Синтаксис описания класса и создание объекта (экземпляра класса) в среде С#.										
Принцип инкапсуляции в ООП. Понятие инкапсуляции. Цель и достоинства инкапсуляции. Технологические средства реализации инкапсуляции в среде программирования С#. Свойства объекта и их реализация.	15	5		2		ПР-1	7	8		8
Наследование в ООП. Наследование классов. Цель и достоинства наследования классов. Создание иерархических структур средствами наследования классов. Технологические средства реализации наследования в среде программирования С#.	7	1		2		ПР-1	3	4		4
Конструкторы. Конструктор по умолчанию. Конструкторы, создаваемые пользователем. Использование конструктора по умолчанию при наследовании. Вызов в явном виде конструктора наследуемого класса.	17	1		8		ПР-1	9	8		8
Полиморфизм в ООП. Понятие полиморфизма. Цель и достоинства полиморфизма. Технологические средства реализации полиморфизма в среде программирования С#. Отношения между классами. Виртуальные методы. Абстрактные классы.	6	1		3		ПР-1	4	2		2



[illegible]

<b>Наименование и краткое содержание разделов и тем дисциплины Форма промежуточной аттестации по дисциплине</b>	<b>Всего (часы )</b>	<b>Контактная работа (работа во взаимодействии с преподавателем), часы из них<sup>2</sup></b>					<b>Самостоятельн ая работа обучающегося, часы, из них</b>			
		<b>Лекционные занятия</b>	<b>Семинарские занятия</b>	<b>Практические занятия</b>	<b>Лабораторные занятия</b>	<b>Учебные занятия, направленные на проведение текущего контроля успеваемости (коллоквиумы, практические контрольные занятия и др.)*</b>	<b>Всего</b>	<b>Выполнение домашних заданий</b>	<b>Подготовка рефератов и т.п.</b>	<b>Всего</b>
Итого	108	18		34			52	56		56

8. \*Опрос (ПР-1), Практические работы (ПР-2), Реферат (ПР-3), \*Экзамен (УО-4). Текущий контроль проводится за счет времени, отведенного на аудиторные занятия

## **7 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине и методические указания для обучающихся по освоению дисциплины**

При проведении занятий

- лектору:
  - структурированно излагать материал руководствуясь иерархией изложения «сверху-вниз»;
  - четко формулировать определения базовых понятий, методик, алгоритмов и т.п.;
  - по ходу изложения нового материала, базирующегося либо связанным с ранее изложенным материалом кратко повторять базовые положения ранее изложенного, подключая к этому повтору студентов;
  - приводить конкретные примеры использования на практике теоретического излагаемого материала;
  - устраивать совместно со студентами кратковременные «мозговые штурмы» для решения поставленной задачи, а после проводить аналитический разбор предложенных решений и излагать обоснованно оптимальное решение;
  - использовать при проведении лекций презентации, иллюстрирующие базовые положения и примеры иллюстрирующие излагаемый материал;
- Ведущему семинары:
  - четко ставить практическую задачу;
  - обсуждать с обучаемыми возможные пути решения, достоинства и недостатки предлагаемых решений;
  - предоставлять обучаемым свободу выбора, реализуемого им решения;
  - консультировать и направлять обучаемого при планировании выполнения и непосредственного выполнения работы;
  - обеспечивать регулярный контроль выполнения заданий;
  - закреплять базовые понятия, методики, алгоритмы и т.п. изученные на лекциях;
- студенту:
  - вести конспектирование учебного материала;
  - применять аббревиатуры и сокращение слов с целью ускорения записи;
  - активно задавать преподавателю уточняющие вопросы с целью эффективного усвоения излагаемого материала;
  - самостоятельно просмотреть материал последней лекции, приготовить возникшие вопросы и выяснить их с преподавателем на последующих аудиторных занятиях;
  - самостоятельно осваивать предложенный на лекции материал, используя рекомендованные и самостоятельно найденные источники требуемой информации;
  - использовать конспекты лекций при выполнении практических работ, подготовке к семинарам, опросам, зачетам.

В учебном процессе широко используются активные и интерактивные формы (компьютерных симуляций, ролевые игры, разбор конкретных ситуаций, проведение «мозговых штурмов», выполнение групповых заданий). В сочетании с внеаудиторной работой это способствует формированию и развитию профессиональных навыков обучающихся.

#### Тематика семинарских занятий

№ семинара	Тема семинарского задания
C1	Проект «Вещественные числа, представленные в виде двух целых» Построение базового класса. Конструктор по умолчанию и параметрический. Метод занесения данных. Метод визуализации. Метод нахождения НОД.
C2	Проект «Вещественные числа, представленные в виде двух целых» Методы, реализующие основные арифметические операции.
C3	Проект «Вещественные числа, представленные в виде двух целых» Операторы перегрузки
C4	Проект «Матрицы и операции над матрицами». Конструктор по умолчанию и параметрический. Методы занесения, чтения данных и визуализации.
C5	Проект «Матрицы и операции над матрицами». Методы реализации операторов сложения, умножения матриц и транспонирования матрицы.
C6	Проект «Создание классов с использованием существующих ранее классов. Геометрические фигуры» Наследование. Создание классов круг, треугольник и прямоугольник.
C7	Проект «Создание классов с использованием существующих ранее классов. Геометрические фигуры». Наследование. Создание класса Снеговик.
C8	Проект «Создание классов с использованием существующих ранее классов. Поезд.». Включение. Создание класса «Вагон»
C9	Проект «Создание классов с использованием существующих ранее классов. Поезд.» Включение. Создание класса «Поезд».
C10	Проект «Создание классов с использованием существующих ранее классов. Поезд.». Включение. Виртуальные методы.
C11	Проект «Следящие за мышкой глаза». Наследование. Обработка событий на примере манипуляции с движением мыши. Создание класса глаз.
C12	Проект «Следящие за мышкой глаза». Наследование. Обработка событий на примере манипуляции с движением мыши. Создание класса Лицо.
C13	Проект «Морской бой». Анализ необходимых классов.
C14	Проект «Морской бой». Класс Игровое поле.
C15	Проект «Морской бой». Класс Игрок.
C16	Проект «Морской бой». Класс Игра.

#### Методическое обеспечение инновационных форм учебных занятий

1. **Тренинг** —форма активного обучения, целью которого является передача знаний, развитие компетентности профессионального проведения системного анализа и управления.

В рамках тренинга создаются условия для самостоятельного поиска способов решения поставленных задач в области системного анализа.

2. **Анализ конкретных ситуаций (CASE-STUDY)** – техника обучения, использующая описание реальных ситуаций. Обучающиеся должны исследовать ситуацию, разобраться в сути проблем, предложить возможные решения и выбрать лучшее из них. Основываются на реальном фактическом материале или же приближены к реальной ситуации.

3. **Проектный метод обучения** – это совокупность приемов и способов обучения, при которых студенты с помощью коллективной или индивидуальной деятельности по отбору, распределению и систематизации материала по определенной теме, составляют проект.

#### **Методические указания для самостоятельной работы обучающихся**

**Самостоятельная работа** студентов, предусмотренная учебным планом, выполняется в ходе семестра в форме выполнения домашних заданий. Отдельные темы теоретического курса прорабатываются студентами самостоятельно в соответствии с планом самостоятельной работы и конкретными заданиями преподавателя с учетом индивидуальных особенностей студентов. Основной акцент сделан на завершение проектной деятельности, начатой на семинарских занятиях.

## **8. Применяемые образовательные технологии для различных видов учебных занятий и для контроля освоения обучающимися запланированных результатов обучения**

**Формы работы студентов** в ходе изучения дисциплины предусмотрены лекционные, практические занятия (семинарские), выполнение домашних работ.

В рамках преподавания дисциплины используются инновационные технологии обучения: чтение лекций, анализ конкретных ситуаций (CASE-STUDY), тренинги, проектное обучение на практических занятиях.

В учебном процессе, помимо чтения лекций, которые составляют 36% аудиторных занятий, используются активные и интерактивные формы обучения: обсуждение отдельных разделов дисциплины, выбор оптимального решения поставленной задачи «мозговым штурмом», выполнение практических работ, домашних заданий. В сочетании с внеаудиторной работой это способствует формированию и развитию профессиональных навыков обучающихся.

### Перечень обязательных видов работы студента:

- посещение семинарских занятий;
- ответы на теоретические вопросы на семинаре;
- решение практических задач и заданий на семинаре;
- выполнение домашних работ.

При изучении теоретического курса на лекциях предусматривается изложение некоторого объема материала в виде презентации. Активно используется электронная доска, обеспечивающая возможность выделять логические информационные группы разными цветами, позволяющая изображать процессы по стадиям их развития.

При выполнении некоторых практических занятий студентам предлагается работать в малых группах: учебная группа разбивается на несколько небольших групп – по 2-3 человека. Процесс выполнения заданий осуществляется на основе обмена мнениями и выбора оптимального пути решения. После этого каждая группа самостоятельно и независимо выполняет задание. По завершению выполнения заданий группы сравнивают полученные результаты. Такая форма обучения ориентирована на развитие у обучаемых навыка работы в коллективе.

### **Интерактивные образовательные технологии, используемые в аудиторных занятиях**

<b>Вид</b>	<b>Используемые интерактивные</b>	<b>Количество часов по</b>
------------	-----------------------------------	----------------------------

занятия	образовательные технологии	учебному плану
<p style="text-align: right;"><b>П</b></p> <p><b>практические занятия</b></p>	<p><b>Разбор конкретных ситуаций, поиск и сбор данных в интернет-среде, групповые обсуждения результатов, электронное тестирование знаний</b></p>	<p style="text-align: center;"><b>17</b></p>

## 9. Фонд оценочных средств для промежуточной аттестации по дисциплине

### 9.1 Перечень компетенций выпускников образовательной программы с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования

Полный перечень компетенций выпускников образовательной программы **01.03.02 Прикладная математика и информатика** с указанием результатов обучения (знаний, умений, владений), характеризующих этапы их формирования, описание показателей и критериев оценивания компетенций на различных этапах их формирования приведен в картах компетенций образовательной программы.

Перечень компетенций выпускников образовательной программы **01.03.02 Прикладная математика и информатика**, в формировании которых участвует данная дисциплина представлен в разделе 4 рабочей программы дисциплины. Указание результатов обучения (знаний, умений, владений), характеризующих этапы формирования компетенций, описание показателей и критериев оценивания компетенций на различных этапах их формирования представлено ниже.

### 9.2 Описание шкал оценивания

Критерии оценивания ответов студентов на зачете с оценкой:

Оценка «отлично»	Студент показывает не только высокий уровень теоретических знаний по изучаемой дисциплине, но и видит междисциплинарные связи. Умеет анализировать практические ситуации. Ответ построен логично. Материал излагается четко, ясно, аргументировано. Уместно используется информационный и иллюстративный материал.
Оценка «хорошо»	Студент показывает достаточный уровень теоретических и практических знаний, свободно оперирует категориальным аппаратом. Умеет анализировать практические ситуации, но допускает некоторые погрешности. Ответ построен логично, материал излагается грамотно.
Оценка «удовлетворительно»	Студент показывает знание основного лекционного и практического материала. В ответе не всегда присутствует логика изложения. Студент испытывает затруднения при приведении практических примеров.
Оценка «неудовлетворительно»	Студент показывает слабый уровень теоретических знаний, не может привести примеры из реальной практики. Неуверенно и логически непоследовательно излагает материал. Неправильно отвечает на дополнительные вопросы или затрудняется с ответом на них.

Описание шкал оценивания для различных заданий, выполняемых в рамках текущего контроля, представлено в методических материалах, определяющих процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.



### 9.3 Критерии и процедуры оценивания результатов обучения по дисциплине, характеризующих этапы формирования компетенций

**Компетенция ОПК-3** – способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным требованиям

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ по дисциплине (модулю) ШКАЛА оценивания (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		1	2	3	4	5	
31(ОПК-3) Знать: основные парадигмы и языки программирова ния	базов ый	Н е знает (не ориенти руется)	Демонстр ирует фрагментарные знания, допускает грубые ошибки	Дем онстрирует общие, но не структурир ованные знания	Демонстрируе т сформированные, но содержащие отдельные пробелы знания	Демонстрир ует высокий уровень знаний	Опросы, зачет с оценкой
33(ОПК-3) Знать: Знать базовые структуры данных и основные алгоритмы их обработки	базов ый	Н е знает (не ориенти руется)	Демонстр ирует фрагментарные знания, допускает грубые ошибки	Дем онстрирует общие, но не структурир ованные знания	Демонстрируе т сформированные, но содержащие отдельные пробелы знания	Демонстрир ует высокий уровень знаний	Опросы, зачет с оценкой

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
34(ОПК-3) Знать: основы создания компонент программного обеспечения	базовый	Не знает (не ориентируется)	Демонстрирует фрагментарные знания, допускает грубые ошибки	Демонстрирует общие, но не структурированные знания	Демонстрирует сформированные, но содержащие отдельные пробелы знания	Демонстрирует высокий уровень знаний	Опросы, зачет с оценкой
У3(ОПК-3) Уметь: выбирать алгоритмы и выполнять их программную реализацию для решения типовых задач предметной области	базовый	Не умеет	Демонстрирует отдельные умения, допускает грубые ошибки	Демонстрирует общие умения, допускает существенные ошибки	Демонстрирует сформированные умения, допускает несущественные ошибки	Демонстрирует высокий уровень умений	Выполнение практических заданий в течение семестра
В1(ОПК-3) Владеть: современными интегрированными средами разработки	базовый	Не владеет навыками	Демонстрирует отдельные навыки, допускает грубые ошибки	Демонстрирует общие навыки, допускает существенные	Демонстрирует сформированные навыки, допускает несущественные ошибки	Демонстрирует высокий уровень владения навыками	Выполнение практических заданий в течение семестра

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
программного обеспечения				ные ошибки			
В2(ОПК-3) Владеть: навыками тестирования, отладки и верификации программ	базовый	Не владеет навыками	Демонстрирует отдельные навыки, допускает грубые ошибки	Демонстрирует общие навыки, допускает существенные ошибки	Демонстрирует сформированные навыки, допускает несущественные ошибки	Демонстрирует высокий уровень владения навыками	Выполнение практических заданий в течение семестра

**Компетенция** ОПК-4 – способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры  
с применением информационно-коммуникационных технологий  
и с учетом основных требований информационной безопасности

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
		1	2	3	4	5	

РЕЗУЛЬТАТ ОБУЧЕНИЯ по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
34(ОПК-4) Знать: Базовые знания математики, информатики, программирования и информационных технологий	базовый	Не знает (не ориентируется)	Демонстрирует фрагментарные знания, допускает грубые ошибки	Демонстрирует общие, но не структурированные знания	Демонстрирует сформированные, но содержащие отдельные пробелы знания	Демонстрирует высокий уровень знаний	Опросы, зачет с оценкой

**Компетенция ПК-2** – способностью понимать, совершенствовать и применять современный математический аппарат

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
		1	2	3	4	5	
В1(ПК-2) Владеть: кодированием на алгоритмическ их языках	базов ый	Н е владеет навыкам и	Демонстр ирует отдельные навыки, допускает грубые ошибки	Дем онстрирует общие навыки, допускает существен	Демонстрируе т сформированные навыки, допускает несущественные ошибки	Демонстрир ует высокий уровень владения навыками	Выполнен ие практических заданий в течение семестра

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  <i>(критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)</i>					
				ные ошибки			

**Компетенция ПК-17** – способностью к разработке и применению алгоритмических и программных решений, математических методов обработки данных в области профессиональной деятельности

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
		1	2	3	4	5	
31(ПК-17) Знать: формальные методы и инструменты разработки программного продукта	базов  ый	Н  е знает (не ориенти руется)	Демонстр  ирует фрагментарные знания, допускает грубые ошибки	Дем  онстрирует общие, но не структурир ованные знания	Демонстрируе  т сформированные, но содержащие отдельные пробелы знания	Демонстрир  ует высокий уровень знаний	Опросы,  зачет с оценкой

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
32(ПК-17) Знать: общие принципы разработки программных средств	базовый	Не знает (не ориентируется)	Демонстрирует фрагментарные знания, допускает грубые ошибки	Демонстрирует общие, но не структурированные знания	Демонстрирует сформированные, но содержащие отдельные пробелы знания	Демонстрирует высокий уровень знаний	Опросы, зачет с оценкой
34(ПК-17) Знать: способы оптимизации программного кода**)	базовый	Не знает (не ориентируется)	Демонстрирует фрагментарные знания, допускает грубые ошибки	Демонстрирует общие, но не структурированные знания	Демонстрирует сформированные, но содержащие отдельные пробелы знания	Демонстрирует высокий уровень знаний	Опросы, зачет с оценкой
36(ПК-17) Знать: методы построения и оценки эффективности алгоритмов и их программной реализации	базовый	Не знает (не ориентируется)	Демонстрирует фрагментарные знания, допускает грубые ошибки	Демонстрирует общие, но не структурированные знания	Демонстрирует сформированные, но содержащие отдельные пробелы знания	Демонстрирует высокий уровень знаний	Опросы, зачет с оценкой

РЕЗУЛЬТАТ ОБУЧЕНИЯ  по дисциплине (модулю)	Уровень освоения компетенции	КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТА ОБУЧЕНИЯ					ПРОЦЕДУРЫ ОЦЕНИВАНИЯ
		по дисциплине (модулю)					
		ШКАЛА оценивания  (критерии берутся из соответствующих карт компетенций, шкала оценивания (4 или более шагов) устанавливается в зависимости от того, какая система оценивания (традиционная или балльно-рейтинговая) применяется)					
У6(ПК-17) Уметь: разрабатывать пользовательск ий интерфейс, прикладной программный интерфейс, в соответствии с постановкой задачи *)	базов ый	Н е умеет	Демонстр ирует отдельные умения, допускает грубые ошибки	Дем онстрирует общие умения, допускает существен ные ошибки	Демонстрируе т сформированные умения, допускает несущественные ошибки	Демонстрир ует высокий уровень умений	Выполнен ие практических заданий в течение семестра

## **9.4 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций и (или) для итогового контроля сформированности компетенции**

### **Пример материала для самоподготовки студента к зачету с оценкой**

#### **Базовые понятия ООП**

**Требование:** Дать ТОЛЬКО ОПРЕДЕЛЕНИЕ.

Т.е. представьте, что Вы объясняете термин человеку, который не знает, что это такое. И ваша задача, используя НЕ БОЛЕЕ ТРЁХ предложений внятно и однозначно объяснить ему этот термин. Если сможете уложиться в два или лучше одно предложение – это только плюс.

**Зачем это нужно?** Во-первых, когда вы четко даете определение – это обеспечивает наилучшее понимание его вам самим. Во-вторых, обеспечивает возможность профессионально разговаривать с коллегами. В-третьих, приучает излагать свои мысли кратко и четко. Можно продолжать эти в-...дцатых, но и этого достаточно.

#### **Термины:**

1. Класс
2. Объект
3. Поле
4. Метод
5. Свойство
6. Члены класса
7. Конструктор
8. Деструктор
9. Инкапсуляция
10. Наследование
11. Полиморфизм
12. Включение (агрегация)
13. Формальные и фактические параметры
14. Перегрузка методов
15. Перегрузка операций
16. областями видимости членов класса
17. Исключительные ситуации
18. Интерфейсы
19. Концепция типа данных
20. Глобальные и локальные имена



## **ВОПРОСЫ к зачету с оценкой по курсу**

### ***ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ***

1. Объектно-ориентированное программирование как идеология программирования и как технология. Достоинства и недостатки.
2. Основные понятия объектно-ориентированного программирования — класс, объект, поле, метод, свойство.
3. Класс. Синтаксис объявления класса. Члены класса. Ссылка на созданный объект класса.
4. Конструкторы и деструкторы. Функциональное назначение. Виды конструкторов.
5. Объекты и их жизненный цикл. Создание и уничтожение объектов. Алгоритм реализации в .Net освобождения памяти.
6. Инкапсуляция. Определение. Функциональное назначение. Реализация. Примеры применения.
7. Инкапсуляция. Свойства. Функциональное назначение. Реализация. Создание индексатора. Примеры применения.
8. Инкапсуляция. Скрытие членов класса. Функциональное назначение. Реализация. Примеры применения.
9. Способы построения новых классов из существующих ранее. Наследование. Функциональное назначение. Реализация. Примеры применения.
10. Наследование. Конструктор по умолчанию. Назначение. Вызов в явном виде конструктора наследуемого класса.
11. Наследование. Проблема тождественности имен членов классов. Реализация. Примеры применения.
12. Способы построения новых классов из существующих ранее. Агрегация (включение). Функциональное назначение. Реализация. Примеры применения. Отличия от наследования. Жизненный цикл агрегируемых объектов.
13. Методы. Определение. Функциональное назначение. Способы реализации. Примеры применения.
14. Полиморфизм. Функциональное назначение. Способы реализации. Примеры применения.
15. Перегрузка методов. Функциональное назначение. Способ реализации. Примеры применения.
16. Виртуальные методы. Функциональное назначение. Примеры применения.
17. Перегрузка операций. Функциональное назначение. Способ реализации. Примеры применения.
18. Управление областями видимости членов класса. Функциональное назначение. Способ реализации. Примеры применения.
19. Исключительные ситуации. Понятие. Способы обработки исключительных ситуаций. Примеры применения.
20. Интерфейсы. Функциональное назначение. Иерархия интерфейсов. Множественное наследование: проблемы и способы их разрешения.
21. Структуры (struct) и перечисления (enum). Отличия структур от классов.

## *СИНТАКСИС ЯЗЫКА*

1. Концепция типа данных. Встроенные (Build-in) типы данных и их реализация в языке C#.
  2. Концепция типа данных. Соглашения о совместимости и приведение типов.
  3. Концепция типа данных. Числовые типы данных.
  4. Концепция типа данных. Символьные типы данных.
  5. Концепция типа данных. Составные типы данных. Массивы и их реализация в C#. Структуры.
  6. Концепция типа данных. Явное и неявное преобразование типов.
  7. Концепция типа данных. Определение собственных типов данных.
  8. Концепция типа данных. Значение (размерные) (Value type) и ссылочные (Reference type) типы данных. Упаковка и распаковка (Boxing, Unboxing).
  9. Концепция типа данных. Переменные и константы и их реализация в C#.
  10. Принцип модульности программ. Понятие глобальных и локальных имен. Область видимости имен. Выбор области видимости.
  11. Принцип модульности программ. Метод, как отдельный модуль программы. Интерфейсная и скрытая часть метода. Формальные и фактические параметры метода. Примеры применения.
  12. Унарные и мультипликативные операции. Примеры применений.
  13. Аддитивные и сдвиговые операции. Примеры применений.
  14. Операции отношения и действий над типами данных. Примеры применений.
  15. Логические операции. Примеры применений.
  16. Организация циклов в C#. Примеры применений.
  17. Операторы перехода и оператор присваивания.
  18. Операторы условного перехода. Примеры применений.
- 
1. Способы построения новых классов из существующих ранее. Агрегация (включение). Функциональное назначение. Реализация. Примеры применения. Отличия от наследования. Жизненный цикл агрегируемых объектов.
  2. Операторы условного перехода. Примеры применений.

## **9.5 Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.**

Для контроля знаний студентов по данной дисциплине необходимо проводить текущий и промежуточный контроль. Текущий контроль выполняется в виде приема допусков и защит практических и индивидуальных работ, устного опроса на лекциях и практических занятиях. Промежуточный контроль проводится в виде учета выполняемой в семестре работы на основе балльно-рейтинговой системы. При рейтинговой системе все знания, умения и навыки, приобретаемые студентами в результате изучения дисциплины, оцениваются в баллах.

Оценка качества работы в семестре в рейтинговой системе является накопительной и используется для оценивания системной работы студентов в течение всего периода обучения. Максимально возможная сумма баллов по всем видам заданий приравнивается 100. Учитывается посещаемость занятий (максимально 30 баллов). Каждое практическое задание оценивается в баллах в зависимости от сложности от его сложности. Максимальный балл за каждый вид задания определяется преподавателем в зависимости от уровня сложности задания в начале семестра и доводится до сведения студентов.

Чтобы быть допущенным к зачету, студент обязан проделать и сдать все практические работы. Работы, пропущенные без уважительной причины, а также работы, не защищенные студентом в течение 2-х недель после проведения практической части, оцениваются преподавателем с понижающим коэффициентом 0,5.

Результаты работы студентов фиксируются преподавателем в электронном журнале успеваемости и размещаются на ресурсе доступном студентам.

## **10.Ресурсное обеспечение**

### **10.1 Перечень основной и дополнительной учебной литературы**

#### **Основная литература**

1. Вайсфельд М. Объектно-ориентированное мышление / Вайсфельд Мэтт. - СПб. : Питер, 2014. - 304с. : ил. - (Библиотека программиста). - ISBN 9785496007931.
2. Биллиг В.А. Основы объектного программирования на C# (C# 3.0, Visual Studio 2008) : Учебное пособие / Биллиг Владимир Арнольдович. - М. : БИНОМ. Лаборатория знаний : ИНТУИТ.РУ, 2013. - 582с. : ил. - (Основы информационных технологий). - ISBN 9785996302598.
3. Шилдт, Г. (Schildt H.). Полный справочник по C# / Г. Шилдт; Зав.ред. С.Н.Тригуб; Пер.с англ.,ред. Н.М.Ручко. - М. : Вильямс, 2008. - 752с. - Прил.:с.731-739.- Предм.указ.:с.740. - ISBN 978-5-8459-0563-5.

### **Дополнительная литература**

1. Фаронов В.В. Программирование на языке С# : Учебный курс / В. В. Фаронов; Зав.ред. А.Кривцов. - СПб. : Питер, 2007. - 240с. : ил. - ISBN 978-5-91180-369-8. - ISBN 5-91180-369-0.
2. Павловская Т.А. С#. Программирование на языке высокого уровня : Учебник для вузов / Т. А. Павловская; Рец. Н.Н.Смирнова, В.В.Трофимов. - СПб. : Питер, 2013. - 432с. : ил. - (Учебники для вузов). - Список лит.:с.425-426.-Алф.указ.:с.427-432. - ISBN 9785459010480.
3. Семакин И.Г. Основы алгоритмизации и программирования : Учебник / Семакин Игорь Геннадьевич, Шестаков Александр Петрович ; Рец. Е.И.Ночка, А.А.Соломашкин. - 2-е изд.,стер. - М. : Академия, 2014. - 304с. - (Профессиональное образование: Информатика и вычислительная техника). - Список лит.:с.138-139. - ISBN 9785446814084.

### **10.2 Периодические издания**

1. Вестник Международного университета природы, общества и человека "Дубна" / гл. ред. Е.Н. Черемисина. – Дубна: Международный университет природы, общества и человека "Дубна".- (Системный анализ в современном обществе). –Журнал.
2. Программные продукты и системы: научно-практическое издание. / гл. ред. С.В. Емельянов. – Тверь: МНИИПУ. – Журнал. – Международное научно-практическое приложение к журналу "Проблемы теории и практики управления".

### **10.3 Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

#### **Электронно-библиотечные системы и базы данных научного содержания**

1. Электронно-библиотечная система «Znaniy.com»: <http://znaniy.com/>.
2. Электронно-библиотечная система «Университетская библиотека онлайн»: [www.bibloclub.ru](http://www.bibloclub.ru)
3. Научная электронная библиотека (НЭБ): <http://elibrary.ru/defaultx.asp>
4. ООО «Издательство Лань», «Лань»: Раздел «Инженерные науки» издательств Машиностроение, ЭНАС, Лань: <http://e.lanbook.com/>
5. БД российских журналов East View : <http://dlib.eastview.com>
6. Базы данных компании EBSCO Publishing: <http://search.ebscohost.com/>
7. БД российских научных журналов на Elibrary.ru (РУНЭБ): <http://elibrary.ru/defaultx.asp>.

#### **Профессиональные ресурсы сети «Интернет»**

1. Национальный открытый университет «Интуит» <http://www.intuit.ru>.

#### **10.4 Перечень информационных технологий, используемых при осуществлении образовательного процесса, включая программное обеспечение, информационные справочные системы (при необходимости)**

Используется лицензионное программное обеспечение:

- Visual Studio (свободно распространяемое ПО)
- MS Power Point
- MS Word Office
- Браузер Интернет

### **11. Описание материально-технической базы**

Лекционные занятия.

Специальные помещения: Учебная аудитория для проведения занятий лекционного и семинарского

типа, групповых и индивидуальных консультаций, текущего контроля.

Аудитория укомплектована: специальной (учебной мебелью) на 134 посадочных места, доской, интерактивной тумбой, интерактивной доской, проекторами, радио системой, акустической системой, усилителем мощности для систем трансляции оповещения.

Практические занятия/семинары.

Специальные помещения: Компьютерный класс (учебная аудитория) для групповых и индивидуальных консультаций, организаций самостоятельной работы, в том числе, научно исследовательской. Аудитория укомплектована: специальной (учебной мебелью) на 12 посадочных

Специализированный компьютерный класс подключенный к сети Интернет и к локальной сети университета, обеспечивающей доступ к программному обеспечению и ЛМС системы MOODLE для проведения семинарских занятий.

Обучающиеся из числа инвалидов и лиц с ограниченными возможностями здоровья могут использовать специализированное программное и материально-техническое обеспечение:

- обучающиеся с нарушениями опорно-двигательного аппарата при необходимости могут использовать адаптивные технические средства: функцию «сенсорная клавиатура», «управление указателем мыши с клавиатуры», специально оборудованные джойстики, увеличенные выносные кнопки, клавиатуры с большими клавишами или накладки «Клавиата»;

- обучающиеся с ограничениями по зрению могут прослушать доступный аудиоматериал или прочитать тексты, увеличив шрифт на экране монитора компьютера.

Рекомендуется использовать экранную лупу и другие визуальные вспомогательные средства, чтобы изменить шрифт текста, межстрочный интервал, синхронизацию с речью и т.д., программы экранного доступа (скринридеры для прочтения текстовой информации через синтезированную речь) и/или включить функцию «экранного диктора» на персональном компьютере с операционной системой Windows 7, 8, 10, Vista, XP.

Студенты с полным отсутствием зрения могут использовать тексты, напечатанные шрифтом Брайля, а для набора текста на компьютере – клавиатуры Брайля;

- обучающиеся с ограничениями по слуху могут воспользоваться индивидуальными техническими средствами (аппараты «Глобус», «Монолог», индивидуальными слуховыми аппаратами, компьютерной аудиогарнитурой, наушниками и др.) при прослушивании необходимой информации, а также услугами сурдопереводчика.

При необходимости обучающиеся с ограниченными возможностями здоровья и инвалиды обеспечиваются печатными и (или) электронными образовательными ресурсами (образовательная программа, учебники, учебные пособия и др.) в формах,

адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,
- в форме аудиофайла,
- в печатной форме на языке Брайля.

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,
- в форме аудиофайла).

Полное описание аудиторий для реализации образовательной программы по данной дисциплине представлено в Справке материально-технического обеспечения текущего года обучения.

## **12 Язык преподавания**

Русский язык