

Рабочая программа дисциплины: разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования (ФГОС ВО 3++) по направлению подготовки 01.03.04 Прикладная математика, утвержденного приказом МИНОБРНАУКИ РФ от 10 января 2018 года № 11 на основании учебного плана, принятого УС ДПИ НГТУ

протокол от 05.06.2024 № 10

Рабочая программа одобрена на заседании кафедры-разработчика РПД Автоматизация, энергетика, математика и информационные системы
протокол от 10.06.2024 № 7

Зав. кафедрой к.т.н, доцент _____ Л.Ю. Вадова
(подпись)

СОГЛАСОВАНО:

Заведующий выпускающей кафедрой Автоматизация, энергетика, математика и информационные системы
к.т.н, доцент _____ Л.Ю. Вадова
(подпись)

Начальник ОУМБО _____ И.В. Старикова
(подпись)

Рабочая программа зарегистрирована в ОУМБО: 01.03.04 - 34

СОДЕРЖАНИЕ

1.	Цели и задачи освоения дисциплины	4
2.	Место дисциплины в структуре образовательной программы	4
3.	Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля)	4
4.	Структура и содержание дисциплины	6
5.	Текущий контроль успеваемости и промежуточная аттестация по итогам освоения дисциплины	10
6.	Учебно-методическое обеспечение дисциплины	14
7.	Информационное обеспечение дисциплины	15
8.	Образовательные ресурсы для инвалидов и лиц с ОВЗ	16
9.	Материально-техническое обеспечение, необходимое для осуществления образовательного процесса по дисциплине	17
10.	Методические рекомендации обучающимся по освоению дисциплины	18
11.	Оценочные средства для контроля освоения дисциплины	20

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

1.1. Цель освоения дисциплины:

Целью освоения дисциплины является изучение подходов к проектированию архитектуры программного средства и принципов разработки программного обеспечения.

1.2. Задачи освоения дисциплины (модуля):

- применение базовых методов и алгоритмов разработки программного продукта;
- знание современных парадигм, технологий и языков программирования и приемов проектирования архитектуры программного средства

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Учебная дисциплина Базы данных включена в перечень дисциплин вариативной части (формируемой участниками образовательных отношений), определяющий направленность ОП. Дисциплина реализуется в соответствии с требованиями ФГОС, ОП ВО и УП.

Дисциплина базируется на следующих дисциплинах: Информатика, Программирование для ЭВМ, Объектно-ориентированное программирование.

Дисциплина Технологии программирования является основополагающей для изучения следующих дисциплин: Алгоритмы и структуры данных, Программирование графических приложений, Численные методы, Многопоточное программирование, Формальные языки и теория компиляции.

Рабочая программа дисциплины для инвалидов и лиц с ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся, по их личному заявлению.

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

Таблица 1 – Формирование компетенции ПКС-1 дисциплинами

Наименование дисциплин, формирующих компетенцию совместно	Семестры формирования компетенций дисциплинами.							
	1	2	3	4	5	6	7	8
Объектно-ориентированное программирование								
Технологии программирования								
Ознакомительная практика								
Проектно-технологическая практика								
Многопоточное программирование								
Программирование для Интернет								
Архитектура ЭВМ и системное ПО								
Преддипломная практика								
Выполнение и защита выпускной квалификационной работы								

ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОП

Таблица 2

Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине			Оценочные средства	
					Текущего контроля	Промежуточной аттестации
ПКС-1. Способен проектировать и разрабатывать компоненты программного обеспечения на основе современных парадигм, технологий и языков программирования.	ИПКС-1.2 - Выполняет разбиение предметной области на объекты и реализацию этих объектов и их взаимосвязей в виде программы.	Знать: подходы к проектированию архитектуры программного средства; принципы разработки программного модуля; служебные подпрограммы, применяемые для отладки и тестирования приложений; источники ошибок в программном средстве; основные пути борьбы с ошибками.	Уметь: создавать надежные программные продукты с использованием современных парадигм, технологий и языков программирования; проектировать архитектуру программного средства.	Владеть: базовыми методами и алгоритмами разработки программного продукта; практическими навыками проектирования и реализации программных средств с использованием объектно-ориентированного языка программирования.	Тестирование в системе MOODLE. (2 тестирования, в базе каждого тестирования 100-110 вопросов), выполнение 1 контрольной работы (25 вариантов)	Вопросы для устного собеседования: билеты (20 билетов)

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1 Распределение трудоёмкости дисциплины по видам работ по семестрам

Общая трудоёмкость дисциплины составляет 2 зач.ед./72 часа, распределение часов по видам работ семестрам представлено в табл. 3.

Формат изучения дисциплины: с использованием элементов электронного обучения

Таблица 3

Распределение трудоёмкости дисциплины по видам работ по семестрам для студентов очной формы обучения

Вид учебной работы	Всего часов	Семестр
		3
1. Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего), в том числе:	38	38
1.1. Аудиторные занятия (всего), в том числе:	34	34
- лекции (Л)	17	17
- лабораторные работы (ЛР)	17	17
- практические занятия (ПЗ)		
- практикумы (П)		
1.2. Внеаудиторные занятия (всего), в том числе:	4	4
- групповые консультации по дисциплине	2	2
- групповые консультации по промежуточной аттестации (зачет с оценкой)	2	2
2. Самостоятельная работа студента (СРС) (всего)	34	34
Вид промежуточной аттестации зачет с оценкой		
Общая трудоемкость, часы/зачетные единицы	72/2	72/2

4.2 Содержание дисциплины, структурированное по темам

Содержание дисциплины, структурированное по темам, приведено в таблице 4.

В столбце «Вид СР» введены следующие сокращения:

«Лекции» – предполагает изучение материалов учебников и учебных пособий для подготовки к лекциям и повторение материала после прослушивания лекции для участия в обсуждениях на практических занятиях.

«Практика» - предполагает использование методических разработок для помощи при решении индивидуальных задач и решение задач из задачников.

Таблица 4

Содержание дисциплины, структурированное по темам, для студентов очной формы обучения

Планируемые (контролируемые) результаты освоения: код УК; ОПК; ПК и индикаторы достижения компетенций	Наименование тем	Виды учебной работы				Вид СРС	Наименование используемых активных и интерактивных образовательных технологий	Реализация в рамках практической подготовки (трудоемкость в часах)	Наименование разработанного электронного курса (трудоемкость в часах)
		Контактная работа			Самостоятельная работа обучающихся (СРС), час				
		Лекции, час	Лабораторные работы, час	Практические занятия, час					
3 семестр									
ПКС-1, ИПКС- 1.2	Тема 1 Ведение в технологию программирования	1	-	-	2	Подготовка к лекциям, тестированию, выполнение заданий для самостоятельной работы. 6.1.1: С. 6-11, 11-16, 52-67, 112-115, 130-133, 167-170. 6.1.2: 14-37, 37-54, 137-164, 54-72	Тестирование в системе MOODLE		
	Тема 2 Жизненный цикл программного средства	2	-	-	4				
	Тема 3 Методы проектирования программного обеспечения	2	-	-	4				
	Тема 4 Технологии коллективной разработки программного обеспечения	2	-	-	4			Подготовка к лекциям, тестированию, выполнение заданий для самостоятельной работы. 6.1.3: С. 306-323, 6.2.1: С. 5-19	

	Тема 5 Технологии коллективной разработки программного обеспечения	2	-		4	Подготовка к лекциям, тестированию, выполнение заданий для самостоятельной работы. 6.1.3:С. 306-323, 6.2.1: С. 5-19	Собеседование		
	Тема 6 Технология структурного программирования	2	8		4	Подготовка отчета о лабораторной работе, подготовка к собеседованию при сдаче лабораторной работы. 6.1.36 С.348-375, 6.2.1: С. 19-39	Собеседование		
	Тема 7 Технология объектно-ориентированного программирования	2	9		4	Подготовка отчета о лабораторной работе, подготовка к собеседованию при сдаче лабораторной работы 6.1.3: 402-415, 6.2.1: С. 39-49	Собеседование		
	Тема 8 Документирование и оценка качества программных продуктов	2			4	Подготовка к лекциям, тестированию, выполнение заданий для самостоятельной работы 6.1.36 С.326-332, 335, 6.2.1: С. 50-59	Собеседование		

	Тема 9 Испытания и сертификация программных средств	2	-	-	4	Подготовка к лекциям, тестированию, выполнение заданий для самостоятельной работы. 6.1.1: 23-29, 29-41, 41-50, 81-89, 90-100. 6.1.2: 72-90, 90-124, 170-173, 125-137	Тестирование в системе MOODLE		
ИТОГО по дисциплине		17	17	-	34		Зачет с оценкой		

5. ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ.

5.1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений и навыков и (или) опыта деятельности

1) Вопросы для собеседования при сдаче отчетов по лабораторным работам (пример).

Лабораторная работа «ООП. Классы»

1. Объектно-ориентированный анализ и проектирование
2. Объектно-ориентированное программирование. Основные понятия
3. Понятие класса в ООП. Описание класса.
4. Стандартная библиотека классов
5. Потокосные классы
6. Контейнерные классы
7. Основные виды методов класса.
8. Реализация класса в C++. Отделение интерфейса от реализации
9. Создание объектов класса. Доступ к полям и методам класса.
10. Дружественные функции и классы.
11. Наследование классов. Множественное наследование
12. Шаблон класса
13. Перегрузка методов
14. Перегрузка операций
15. Виртуальные функции. Виртуальные классы.
16. Оператор `this`.
17. Пространство имен
18. Класс памяти и область действия переменных
19. Потокосный ввод-вывод библиотек `<iostream.h` и `stdio.h`
20. Динамическая и статическая память

2) Пример задания для самостоятельной работы

Задание на классы.

1. Определить пользовательский класс в соответствии с вариантом задания.
2. Определить в классе следующие конструкторы: без параметров, с параметрами, копирования.
3. Определить в классе деструктор.
4. Определить в классе методы для просмотра и установки полей данных с проверкой корректности ввода.
5. Определить в классе дополнительно два (или более 2) собственных метода.
6. Рассмотреть различные способы доступа к открытым элементам класса: через объект, ссылку, указатель.
7. Написать демонстрационную программу, в которой создаются и разрушаются объекты пользовательского класса и каждый вызов конструктора и деструктора сопровождается выдачей соответствующего сообщения (какой объект какой конструктор или деструктор вызвал).
8. Следует предусмотреть в программе все возможные способы вызова конструктора копирования.
9. В программе необходимо предусмотреть размещение объектов как в статической, так и в динамической памяти, а также создание массивов объектов.
10. Программа должна быть реализована, как минимум, тремя файлами.

3) **Тесты для текущего и промежуточного контроля знаний обучающихся**
Тесты, проводимые на электронной платформе Moodle на сайте ДПИ НГТУ по адресу: <http://dpingtu.ru/Moodle>.

Включают решение задач по темам курса с выбором правильного варианта ответа.

4) **Перечень вопросов выносимых на промежуточные аттестации**

ТЕОРЕТИЧЕСКИЕ ВОПРОСЫ к зачету с оценкой

1. Технология программирования, понятие определение. «Стихийное» программирование.
2. Технология программирования, понятие определение. Структурный подход к программированию.
3. Технология программирования, понятие определение. Объектный подход к программированию,
4. Технология программирования, понятие определение. Компонентный подход к программированию и CASE-технологии.
5. Жизненный цикл ПО. Структура ЖЦ ПО.
6. Модели жизненного цикла программных средств. Водопадный подход.
7. Качество ПО. Эволюция понятия качества ПО.
8. Критерии качества ПО: Функциональность, Надежность, Практичность. Их атрибуты.
9. Критерии качества ПО: Эффективность, Сопровождаемость, Мобильность. Их атрибуты.
10. Модули. Модульное программирование.
11. Сцепление модулей.
12. Связность модулей.
13. Методы разработки структуры ПС. Восходящая разработка ПС.
14. Нисходящая разработка ПС
15. Средства описания структурных алгоритмов.
16. Тестирование ПО. Информационные потоки процесса тестирования.
17. Функциональное тестирование. методы формирования тестовых наборов.
18. Структурное тестирование. Особенности тестирования.
19. Разработка пользовательских интерфейсов. Разработка примитивного интерфейса и интерфейса-меню.
20. Разработка пользовательских интерфейсов. Разработка интерфейсов со свободной навигацией и прямого манипулирования. Интеллектуальные элементы пользовательских интерфейсов.
21. Разработка пользовательских интерфейсов. Этапы разработки пользовательского интерфейса.
22. Пользовательская и программная модели интерфейса. Критерии оценки интерфейса пользователем.
23. Классификация ошибок ПО.
24. Методы отладки программного обеспечения.
25. Разработка программной документации. Виды программных документов (ЕСПД).
26. Разработка программной документации. Руководство пользователя. Руководство системного программиста.
27. Принципиальные решения начальных этапов проектирования. Выбор архитектуры программного обеспечения.
28. Принципиальные решения начальных этапов проектирования. Выбор типа пользовательского интерфейса. Выбор подхода к разработке.

29. Принципиальные решения начальных этапов проектирования. Выбор языка программирования. Выбор или формирование стандартов разработки.
30. Разработка структурной и функциональной схем
31. Стиль программирования. Стиль оформления текстов модулей.
32. Технология программирования и основные этапы ее развития.
33. Проблемы разработки сложных программных систем.
34. Жизненный цикл и этапы разработки программного обеспечения.
35. Оценка качества процессов создания программного обеспечения.
36. Понятие технологичности программного обеспечения.
37. Классификация программных продуктов по функциональному признаку.
38. Основные эксплуатационные требования к программным продуктам.
39. Принципиальные решения начальных этапов проектирования.
40. Спецификации программного обеспечения при структурном подходе.
41. Математические модели задач, разработка или выбор методов решения.
42. Анализ требований и определение спецификаций программного обеспечения при объектном подходе.
43. Проектирование программного обеспечения при объектном подходе.
44. Разработка пользовательских интерфейсов. Типы пользовательских интерфейсов и этапы их разработки.
45. Пользовательская и программная модели интерфейса.
46. Классификации диалогов и общие принципы их разработки.
47. Тестирование программных продуктов.
48. Отладка программного обеспечения

5.2. Описание показателей и критериев контроля успеваемости, описание шкал оценивания

Для оценки знаний, умений, навыков и формирования компетенции по дисциплине применяется балльно-рейтинговая система контроля и оценки успеваемости обучающихся очной формы и традиционная система контроля и оценки успеваемости обучающихся заочной формы. Основные требования балльно-рейтинговой системы по дисциплине и шкала оценивания приведены в таблицах 6-8.

Таблица 6

Требования балльно-рейтинговой системы по дисциплине

Виды работ	Количество подвидов в работы	Максимальные баллы за подвид работы				Сроки выполнения	Дополнительные баллы за своевр. выполн.	Дополнительные баллы за качество	Штрафные баллы	
		1	2	3	4				За нарушение сроков	За качество
Тестирование	4	5	5	5	5	ежемесячно				
Выполнение лабораторных работ	8	По 3 баллов за 1 работу				еженедельно		До +2 за 1 работу	До -2 за 1 работу	До -2 за 1 работу
Выполнение дополнительных д/з повышенной сложности (для желающих)	4 *	По 5 баллов за 1 работу						До +5 баллов за 1 работу		
Посещение занятий (участие в обсуждениях задач)	17	1 балл за одно занятие				еженедельно			По -1 баллу за 1 пропуск	

Таблица 7

Критерии оценивания результата обучения по дисциплине и шкала оценивания

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Критерии оценивания результатов обучения			
		Оценка «неудовлетворительно» / «не зачтено» 0-54% от тах рейтинговой оценки контроля	Оценка «удовлетворительно» / «зачтено» 55-70% от тах рейтинговой оценки контроля	Оценка «хорошо» / «зачтено» 71-85% от тах рейтинговой оценки контроля	Оценка «отлично» / «зачтено» 86-100% от тах рейтинговой оценки контроля
ПКС-1. Способен проектировать и разрабатывать компоненты программного обеспечения на основе современных парадигм, технологий и языков программирования	ИПКС-1.2 - Выполняет разбиение предметной области на объекты и реализацию этих объектов и их взаимосвязей в виде программы	Изложение учебного материала бессистемное, неполное, не знает основ технологий программирования, не может использовать их в рамках поставленных целей и задач, что препятствует усвоению последующего материала	Фрагментарные, поверхностные знания по базовыми методами и алгоритмами разработки программного продукта. Изложение полученных знаний неполное, однако это не препятствует усвоению последующего материала. Допускаются отдельные существенные ошибки, исправленные с помощью преподавателя. Затруднения при формулировании основных положений и их применении	Знает материал на достаточно хорошем уровне; представляет основные задачи в рамках постановки целей и выбора оптимальных способов их достижения. Обладает хорошими практическими навыками проектирования и реализации программных средств с использованием объектно-ориентированного языка программирования	Имеет глубокие знания всего материала структуры дисциплины; освоил новации лекционного курса по сравнению с учебной литературой; изложение полученных знаний полное, системное; допускаются единичные ошибки, самостоятельно исправляемые при собеседовании

Критерии оценивания

Оценка	Критерии оценивания
Высокий уровень «5» (отлично) - зачтено	оценку « отлично » заслуживает обучающийся, освоивший знания, умения, компетенции и теоретический материал без пробелов; выполнивший все задания, предусмотренные учебным планом на высоком качественном уровне; практические навыки профессионального применения освоенных знаний сформированы.
Средний уровень «4» (хорошо) - зачтено	оценку « хорошо » заслуживает обучающийся, практически полностью освоивший знания, умения, компетенции и теоретический материал, учебные задания не оценены максимальным числом баллов, в основном сформировал практические навыки.
Пороговый уровень «3» (удовлетворительно) - зачтено	оценку « удовлетворительно » заслуживает обучающийся, частично с пробелами освоивший знания, умения, компетенции и теоретический материал, многие учебные задания либо не выполнил, либо они оценены числом баллов близким к минимальному, некоторые практические навыки не сформированы.
Минимальный уровень «2» (неудовлетворительно) – не зачтено	оценку « неудовлетворительно » заслуживает обучающийся, не освоивший знания, умения, компетенции и теоретический материал, учебные задания не выполнил, практические навыки не сформированы.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**6.1. Учебная литература**

6.1.1 Орлов С.А. Технологии разработки программного обеспечения: Учебное пособие для вузов.- М.: СПб, 2003.- 317 с.

6.1.2 Капранов С.Н. Основы алгоритмов и структур данных [Электронные текстовые данные]: Учебное пособие для вузов.- Н.Новгород, НГТУ, 2012. URL: http://fdp.nntu.ru/books/Osnovy_algoritmov_i_structur_dannih/files/assets/basic-html/index.html#page16.

6.1.3 Павловская Т.А. Си/C++ Программирование на языке высокого уровня: Учебник для вузов.- СПб: Питер, 2007.- 426 с.

6.1.4 Павловская Т.А. Си/C++ Структурное и объектно-ориентированное программирование: Учебное пособие для вузов.- СПб: Питер, 2008.- 352 с.

Библиотечный фонд укомплектован печатными изданиями из расчета не менее 0,25 экземпляра каждого из изданий, указанных выше на каждого обучающегося из числа лиц, одновременно осваивающих соответствующую дисциплину (модуль).

6.2. Методические указания, рекомендации и другие материалы к занятиям

6.2.1 Объектно-ориентированное программирование: Методические указания к выполнению лабораторных работ по дисциплине «Технология программирования» / Сост. Н.О. Кулигина –Дзержинск, 2015

6.2.2 Классы и объекты С++: Методические указания к лабораторной работе №1 по дисциплине «Технология программирования» / Сост. Н.О. Кулигина –Дзержинск, 2016

6.2.3 Шаблоны классов: Методические указания к лабораторной работе №2 по дисциплине «Технология программирования» / Сост. Н.О. Кулигина –Дзержинск, 2016

6.2.4 Перегрузка операций: Методические указания к лабораторной работе №3 по дисциплине «Технология программирования» / Сост. Н.О. Кулигина –Дзержинск, 2016

6.2.5 Наследование и виртуальные функции: Методические указания к лабораторной

работе №4 по дисциплине «Технология программирования» / Сост. Н.О. Кулигина – Дзержинск, 2016

6.2.6 Стандартная библиотека шаблонов STL: Методические указания к лабораторной работе №5 по дисциплине «Технология программирования» / Сост. Н.О. Кулигина – Дзержинск, 2020.

6.2.7 Методические рекомендации для ППС по организации самостоятельной работы студентов по дисциплине "Технологии программирования" для обучающихся направления подготовки 231300 "Прикладная математика" очной формы обучения; ДПИ НГТУ/ Сост. И.Ю. Харитоновна –Дзержинск, 2013.

6.2.8 Методические рекомендации для ППС по организации аудиторной работы студентов по дисциплине "Технологии программирования" для обучающихся направления подготовки 231300 "Прикладная математика" очной формы обучения; ДПИ НГТУ/ Сост. И.Ю. Харитоновна –Дзержинск, 2013.

7. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Учебный процесс по дисциплине обеспечен необходимым комплектом лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства (состав по дисциплине определен в настоящей РПД и подлежит обновлению при необходимости).

7.1. Перечень информационных справочных систем

Дисциплина, относится к группе дисциплин, в рамках которых предполагается использование информационных технологий как вспомогательного инструмента.

Информационные технологии применяются в следующих направлениях: при подготовке и оформлении отчетов о лабораторных работах, выполнении заданий для самостоятельной работы.

Таблица 9

Перечень электронных библиотечных систем

№	Наименование ЭБС	Ссылка к ЭБС
1	Консультант студента	http://www.studentlibrary.ru/
2	Лань	https://e.lanbook.com/

7.2. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины

Таблица 10

Программное обеспечение

№ п/п	Программное обеспечение, используемое в университете на договорной основе	Программное обеспечение свободного распространения
1	Microsoft Windows 10 (подписка MSDN 700593597, подписка DreamSparkPremium, 19.06.19)	Adobe Acrobat Reader https://acrobat.adobe.com/ru/ru/acrobat/pdf-reader.html
2	Microsoft office 2010 (Лицензия № 49487295 от 19.12.2011)	OpenOffice https://www.openoffice.org/ru/
3	Консультант Плюс	PTC Mathcad Express https://www.mathcad.com/ru

Перечень современных профессиональных баз данных и информационных справочных систем

В таблице 11 указан перечень профессиональных баз данных и информационных справочных систем, к которым обеспечен доступ (удаленный доступ).

Таблица 11

Перечень современных профессиональных баз данных и информационных справочных систем

№ п/п	Наименование профессиональной базы данных, информационно-справочной системы	Доступ к ресурсу (удаленный доступ с указанием ссылки/доступ из локальной сети университета)
1	2	3
1	База данных стандартов и регламентов РОССТАНДАРТ	https://www.gost.ru/portal/gost//home/standarts
2	Перечень профессиональных баз данных и информационных справочных систем	https://cyberpedia.su/21x47c0.html
3	Инструменты и веб-ресурсы для веб-разработки – 100+	https://techblog.sdstudio.top/blog/instrumenty-i-veb-resursy-dlia-veb-razrabotki-100-plus
4	Справочная правовая система «КонсультантПлюс»	доступ из локальной сети

8. ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОВЗ

В таблице 12 указан перечень образовательных ресурсов, имеющих формы, адаптированные к ограничениям их здоровья, а также сведения о наличии специальных технических средств обучения коллективного и индивидуального пользования.

Таблица 12

Образовательные ресурсы для инвалидов и лиц с ОВЗ

№	Перечень образовательных ресурсов, приспособленных для использования инвалидами и лицами с ОВЗ	Сведения о наличии специальных технических средств обучения коллективного и индивидуального пользования
1	ЭБС «Консультант студента»	озвучка книг и увеличение шрифта
2	ЭБС «Лань»	специальное мобильное приложение - синтезатор речи, который воспроизводит тексты книг и меню навигации
3	ЭБС «Юрайт»	версия для слабовидящих

Согласно Федеральному Закону об образовании 273-ФЗ от 29.12.2012 г. ст. 79, п.8 "Профессиональное обучение и профессиональное образование обучающихся с ограниченными возможностями здоровья осуществляются на основе образовательных программ, адаптированных при необходимости для обучения указанных обучающихся". АОП разрабатывается по каждой направленности при наличии заявлений от обучающихся, являющихся инвалидами или лицами с ОВЗ и изъявивших желание об обучении по данному типу образовательных программ.

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ, НЕОБХОДИМОЕ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ

Учебные аудитории для проведения занятий по дисциплине, оснащены оборудованием и техническими средствами обучения.

В таблице 13 перечислены:

- учебные аудитории для проведения учебных занятий, оснащенные оборудованием и техническими средствами обучения;
- помещения для самостоятельной работы обучающихся, которые оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду ДПИ НГТУ.

Таблица 13

Оснащенность аудиторий и помещений для самостоятельной работы обучающихся по дисциплине

№	Наименование аудиторий и помещений для самостоятельной работы	Оснащенность аудиторий помещений и помещений для самостоятельной работы	Перечень лицензионного программного обеспечения. Реквизиты подтверждающего документа
1	1329 Аудитория для лекционных занятий Нижегородская обл., г. Дзержинск, ул. Гайдара, д. 49	Комплект демонстрационного оборудования: ПК, с выходом на мультимедийный проектор, на базе Intel Pentium G4560 3.5 ГГц, 4 Гб ОЗУ, монитор 20' – 1 шт. Мультимедийный проектор Epson- 1 шт; Экран – 1 шт.	
2	1234 Научно-техническая библиотека ДПИ НГТУ, студенческий читальный зал; Нижегородская обл., г. Дзержинск, ул. Гайдара, д. 49	Комплект демонстрационного оборудования: ПК, с выходом на мультимедийный проектор, на базе Intel Pentium G4560 3.5 ГГц, 4 Гб ОЗУ, монитор 20' – 1 шт. Мультимедийный проектор Epson- 1 шт; Экран – 1 шт.; Набор учебно-наглядных пособий	<ul style="list-style-type: none"> • Microsoft Windows 10 Домашняя (поставка с ПК) • LibreOffice 6.1.2.1. (свободное ПО) • Foxit Reader (свободное ПО); • 7-zip для Windows (свободное ПО)
3	1443а компьютерный класс - помещение для СРС, курсового проектирования (выполнения курсовых работ), Нижегородская обл., г. Дзержинск, ул. Гайдара, д. 49	ПК на базе Intel Celeron 2.67 ГГц, 2 Гб ОЗУ, монитор Acer 17' – 4 шт. ПК подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду университета	<ul style="list-style-type: none"> • Microsoft Windows 7 (подписка DreamSpark Premium) • Apache OpenOffice 4.1.8 (свободное ПО); • Mozilla Firefox (свободное ПО); • Adobe Acrobat Reader (свободное ПО); • 7-zip для Windows (свободное ПО); • КонсультантПлюс (ГПД № 0332100025418000079 от 21.12.2018);

10. МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ОБУЧАЮЩИМСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

10.1. Общие методические рекомендации для обучающихся по освоению дисциплины, образовательные технологии

Дисциплина реализуется посредством проведения контактной работы с обучающимися (включая проведение текущего контроля успеваемости), самостоятельной работы обучающихся и промежуточной аттестации.

Контактная работа: аудиторная, внеаудиторная, а также проводится в электронной информационно-образовательной среде университета (далее - ЭИОС).

Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

- балльно-рейтинговая технология оценивания;
- текущий контроль знаний в форме тестирования в среде MOODLE.

При преподавании дисциплины «Технологии программирования», используются современные образовательные технологии, позволяющие повысить активность обучающихся при освоении материала курса и предоставить им возможность эффективно реализовать часы самостоятельной работы.

Весь лекционный материал курса сопровождается компьютерными презентациями, в которых наглядно преподносятся материал различных разделов курса, что дает возможность обсудить материал с обучающимися во время чтения лекций, активировать их деятельность при освоении материала. Материалы лекций в виде слайдов находятся в свободном доступе на в системе MOODLE и могут быть получены до чтения лекций и проработаны обучающимися в ходе самостоятельной работы.

На лекциях, лабораторных занятиях реализуются интерактивные технологии, приветствуются вопросы и обсуждения, используется лично-ориентированный подход, технология работы в малых группах, что позволяет обучающимся проявить себя, получить навыки самостоятельного изучения материала, выровнять уровень знаний в группе.

Все вопросы, возникшие при самостоятельной работе над домашним заданием подробно разбираются на лабораторных занятиях и лекциях. Проводятся индивидуальные и групповые консультации с использованием как встреч с обучающимися, так и современных информационных технологий (электронная почта).

Иницируется активность обучающихся, поощряется задание любых вопросов по материалу, практикуется индивидуальный ответ на вопросы обучающегося, рекомендуются методы успешного самостоятельного усвоения материала в зависимости от уровня его базовой подготовки.

Для оценки знаний, умений, навыков и уровня сформированности компетенции применяется балльно-рейтинговая система контроля и оценки успеваемости обучающихся в процессе текущего контроля.

Промежуточная аттестация проводится третьем семестре в форме зачета с оценкой, с учетом текущей успеваемости.

Результат обучения считается сформированным на повышенном уровне, если теоретическое содержание курса освоено полностью. При устных собеседованиях обучающийся исчерпывающе, последовательно, четко и логически излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, использует в ответе дополнительный материал. Все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, обучающийся способен анализировать полученные результаты, проявляет самостоятельность при выполнении заданий.

Результат обучения считается сформированным на пороговом уровне, если теоретическое содержание курса в основном освоено. При устных собеседованиях обучающийся последовательно излагает учебный материал; при затруднениях способен после наводящих вопросов продолжить обсуждение, справляется с задачами, вопросами и

другими видами заданий, требующих применения знаний; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, обучающийся способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий

Результат обучения считается несформированным, если обучающийся при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет задания, не демонстрирует необходимых умений, качество выполненных заданий не соответствует установленным требованиям, качество их выполнения оценено числом баллов ниже трех по оценочной системе, что соответствует допороговому уровню.

10.2. Методические указания для занятий лекционного типа

Лекционный курс предполагает систематизированное изложение основных вопросов тематического плана. В ходе лекционных занятий раскрываются базовые вопросы в рамках каждой темы дисциплины (таблица 5 и 6). Обозначаются ключевые аспекты тем, а также делаются акценты на наиболее сложные и важные положения изучаемого материала. Материалы лекций являются опорной основой для подготовки обучающихся к практическим занятиям / лабораторным работам и выполнения заданий самостоятельной работы, а также к мероприятиям текущего контроля успеваемости и промежуточной аттестации по дисциплине.

10.3. Методические указания по освоению дисциплины на лабораторных работах

Подготовку к каждой лабораторной работе обучающийся должен начать с ознакомления с планом занятия, который отражает содержание предложенной темы. Каждая выполненная работа с оформленным отчетом подлежит защите у преподавателя.

При оценивании лабораторных работ учитывается следующее:

- качество выполнения экспериментально-практической части работы и степень соответствия результатов работы заданным требованиям;
- качество оформления отчета по работе;
- качество устных ответов на контрольные вопросы при защите работы.

10.4. Методические указания по самостоятельной работе обучающихся

Самостоятельная работа обеспечивает подготовку обучающихся к аудиторным занятиям и мероприятиям текущего контроля и промежуточной аттестации по изучаемой дисциплине. Результаты этой подготовки проявляются в активности обучающихся на занятиях и в качестве выполненных заданий для самостоятельной работы и других форм текущего контроля.

При выполнении заданий для самостоятельной работы рекомендуется проработка материалов лекций по каждой пройденной теме, а также изучение рекомендуемой литературы, представленной в Разделе 6.

В процессе самостоятельной работы при изучении дисциплины обучающиеся могут работать на компьютере в специализированных аудиториях для самостоятельной работы (таблица 13). В аудиториях имеется доступ через информационно-телекоммуникационную сеть «Интернет» к электронной информационно-образовательной среде университета (ЭИОС) и электронной библиотечной системе (ЭБС), где в электронном виде располагаются учебные и учебно-методические материалы, которые могут быть использованы для самостоятельной работы при изучении дисциплины.

11. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ

11.1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта в ходе текущего контроля успеваемости

Для текущего контроля знаний обучающихся по дисциплине проводится комплексная оценка знаний, включающая:

- проведение лабораторных работ;
- выполнение заданий для самостоятельной работы
- тестирование на сайте преподавателя по различным разделам курса

11.1.1. Типовые задания для лабораторных работ

Типовые задания для лабораторных работ приведены в методических указаниях по проведению лабораторных работ (6.2.1).

Вариант 0

Определить массив $A[m][n]$ (m и n вводятся с клавиатуры), заполнить его случайными числами и выполнить следующие вычисления. Результаты выполнения задания выдать на экран и в файл. 1. Сумму положительных элементов массива 2. Произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами 3. Упорядочить элементы по убыванию.

11.1.2. Типовые задания для самостоятельной работы обучающихся очной формы

Задание: построить класс 1-го уровня с указанными в индивидуальном задании полями и методами:

- конструктор;
- функция, которая определяет «качество» объекта – Q по заданной формуле ;
- вывод информации об объекте.

Построить класс 2-го уровня (класс-потомок), который содержит:

- дополнительное поле P ;
- функция, которая определяет «качество» объекта класса 2-го уровня – Q_p , которая перекрывает функцию качества класса 1-го уровня (Q), выполняя вычисление по новой формуле (табл. 11.7, столб 3).

Создать проект для демонстрации работы: ввод и вывод информации об объектах классов 1-го и 2-го уровней.

Варианты индивидуальных заданий (примеры)

№ вар	Поля и функция «качества» (Q) класса 1-го уровня	Поле и функция «качества» Q_p класса 2-го уровня
1	Компьютер: - наименование процессора; - тактовая частота процессора (МГц); - объем оперативной памяти (Мб). $Q = (0,1 \cdot \text{частота}) + \text{память}$	P : объем винчестера (Гб) $Q_p = Q + 0,5 \cdot P$

2	<p>Оператор мобильной связи:</p> <ul style="list-style-type: none"> - название оператора; - стоимость 1 минуты разговора; - площадь покрытия. $Q = 100 \cdot \text{площадь покрытия} / \text{стоимость 1 минуты разговора}$	<p>P: наличие платы за каждое соединение</p> $Qp = 0,7 \cdot Q$, если P - истина, иначе $Qp = 1,5 \cdot Q$
3	<p>Товар на складе:</p> <ul style="list-style-type: none"> - наименование; - цена; - количество. $Q = \text{цена} / \text{количество}$	<p>P: год выпуска товара</p> $Qp = Q + 0,5 \cdot (T - P)$, где T - текущий год
4	<p>Кабель:</p> <ul style="list-style-type: none"> - тип; - количество жил кабеля; - диаметр. $Q = \text{диаметр} / \text{количество жил}$	<p>P: наличие оплетки</p> Qp : если P - истина, то $Qp = 2 \cdot Q$; иначе $Qp = 0,7 \cdot Q$
5	<p>Учебник по программированию:</p> <ul style="list-style-type: none"> - название; - количество страниц; - цена. $Q = \text{цена} / \text{количество страниц}$	<p>P: год издания</p> $Qp = Q - 0,2 \cdot (T - P)$, где T - текущий год

11.1.3. Типовые тестовые задания

1. Можно ли перегрузить функцию `main()`?
 - a) да
 - b) нет

2. В каких случаях необходимо использовать оператор `return` в теле функции?
 - a) Всегда
 - b) если необходимо, чтобы функция вернула значение
 - c) если необходимо обеспечить выход из функции в произвольном месте
 - d) если указан тип возвращаемого значения, в том числе и `void`

3. Что из нижеперечисленного не является прототипом функции?
 - a) `double funct(char x)`
 - b) `char x();`
 - c) `int funct(char x, char y);`
 - d) `void funct();`

4. Что называется прототипом функции?
 - a) описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров
 - b) описание функции, включая ее имя, тип возвращаемого значения, типы параметров
 - c) имя функции и тип возвращаемого значения
 - d) описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров, тело функции

5. Что значит ключевое слово `inline`?

- a) нет правильного ответа
- b) Сообщает компилятору использовать функцию только в пределах одного файла с исходным кодом
- c) все вызовы встроенных функций заменяются кодом этой функции

6. Что означает количество аргументов?

- a) количество аргументов, плюс один
- b) общий размер массива аргументов
- c) количество аргументов
- d) количество аргументов, минус один

7. Дана последовательность чисел: 2, 3, 4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 8. Нумерация элементов начинается с нуля. Элемент с каким номером будет найден методом бинарного поиска по ключу `key=5`?

- a) 9
- b) 11
- c) 8
- d) 7
- e) 3

8. Дан программный код. Какое значение возвращает функция `Search`?

```
int Search(int *x, int k, int key){ int i; for (i = k-1; i >=0 ; i--) if ( x[i] == key ) break; return i > 0 ? i : -1;}
```

- a) номер последнего элемента, совпадающего с ключом поиска
- b) значение минимального элемента массива
- c) номер последнего минимального элемента массива
- d) номер первого элемента, совпадающего с ключом поиска

9. Дан программный код. Какое значение возвращает функция `Search`?

```
int Search(int *x, int k, int key)
{ bool found = false;
  int high = k - 1, low = 0;
  int middle = (high + low) / 2;
  while ( !found && high >= low )
  { if (key == x[middle]) found = true; else if (key < x[middle])
    high = middle - 1;
    else
    low = middle + 1; middle = (high + low) / 2; }
  return found ? middle : -1 ;}
```

- a) номер последнего элемента, совпадающего с ключом поиска
- b) номер первого элемента, совпадающего с ключом поиска
- c) значение среднего элемента массива
- d) номер элемента, совпадающего с ключом поиска

10. Укажите, что возвращает функция, фрагмент кода которой представлен ниже:

```
int f (int k,int x[max]) { int i,m=x[0]; for (i=1;i<k;i++) if (m>x[i]) m=x[i]; return m; }
```

- a) минимальный элемент двумерного массива
- b) минимальный элемент одномерного массива
- c) максимальный элемент одномерного массива
- d) максимальный элемент двумерного массива

11. В программном коде выполнено объявление динамической структуры стека:

```
struct Single_List { int Data; Single_List *Next; };struct Stack { Single_List *Top; }; . . . . .  
. . . . .Stack *Top_Stack;
```

Какое значение содержит Top_Stack->Top?

- a) адрес конца стека
- b) значение элемента из вершины стека
- c) адрес элемента внутри стека
- d) адрес вершины стека

12. Что из себя представляет динамическое выделение памяти?

- a) память под объект (переменную) может выделяться не сразу, а в процессе работы программы, освобождение памяти производится вручную
- b) память под объект (переменную) выделяется каждый раз при обращении к переменной
- c) память под объект (переменную) может выделяться не сразу, а в процессе работы программы, освобождение памяти производится автоматически после завершения программы

11.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта в ходе промежуточной аттестации по дисциплине

11.2.1. Типовые тестовые задания на зачет с оценкой

Вариант 0

Часть А. В каждом задании укажите номер правильного варианта ответа.

1) Что такое объектно-ориентированное программирование?

- А) Методология
- Б) Парадигма
- В) Технология
- Г) Модель

2) Какие принципы являются основными в объектно-ориентированном программировании?

- А) Наследование, полиморфизм
- Б) Полиморфизм, инкапсуляция
- В) Инкапсуляция, наследование, полиморфизм

Г) Агрегация, инкапсуляция

3) Что является пользовательским типом данных, представляющим собой одновременно интерфейс и реализацию?

- А) Класс
- Б) Структура
- В) Перечисление
- Г) Метакласс

4) Что является совокупностью публичных методов данного класса?

- А) Абстракция
- Б) Контракт
- В) Модель
- Г) Интерфейс

5) С помощью какого ключевого слова объявляется класс?

- А) def
- Б) struct
- В) class
- Г) enum

6) Какие модификаторы доступа существуют?

- А) public, protected, private
- Б) opened, protected, closed
- В) public, closed
- Г) opened, private

7) Какие методы класса генерируются автоматически, если их явно не объявить?

- А) Конструктор по умолчанию, деструктор
- Б) Конструктор по умолчанию, копирующий конструктор, деструктор, оператор копирующего присваивания
- В) Конструктор по умолчанию
- Г) Копирующий конструктор, деструктор

8) Выберите корректное объявление копирующего конструктора для класса А.

- А) A(const A&);
- Б) A(A&);
- В) A(const A);
- Г) A(A);

9) В каких случаях вызывается конструктор копирования?

- А) Инициализация объекта другим объектом одного класса, передача объекта в функцию по значению
- Б) Передача объекта в функцию по значению, возврат объекта из функции по значению
- В) Инициализация объекта другим объектом одного класса
- Г) Инициализация объекта другим объектом одного класса, передача объекта в функцию по значению, возврат объекта из функции по значению

10) Какое объявление оператора копирующего присваивания для класса А является корректным?

- А) A operator=(const A);

- Б) A& operator=(const A&);
- В) A& operator=(A);
- Г) A operator=(A&);

Часть В. Решите следующие задания

Написать программу реализующую перегрузку функции для вычисления суммы отрицательных элементов одномерного массива для типов int и float

Регламент проведения текущего контроля в форме компьютерного тестирования

Кол-во заданий в банке вопросов	Кол-во заданий, предъявляемых обучающемуся	Время на тестирование, мин.
220	12	15

Полный фонд оценочных средств для проведения промежуточной аттестации в форме компьютерного тестирования размещен в банке вопросов данного курса дисциплины в СДО MOODLE.