МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)

УТВЕРЖДАЮ
Проректор по УР
_____Сенченко П.В.
«22» _____02_____ 2023 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки / специальность: **09.03.01 Информатика и вычислительная техника**
Направленность (профиль) / специализация: **Программное обеспечение средств вычислительной техники и автоматизированных систем**
Форма обучения: **заочная (в том числе с применением дистанционных образовательных технологий)**
Факультет: **Факультет дистанционного обучения (ФДО)**
Кафедра: **Кафедра автоматизированных систем управления (АСУ)**
Курс: **3, 4**
Семестр: **6, 7**
Учебный план набора 2023 года

Объем дисциплины и виды учебной деятельности

Виды учебной деятельности	6 семестр	7 семестр	Всего	Единицы
Лабораторные занятия	4	4	8	часов
Курсовой проект		4	4	часов
Самостоятельная работа	124	187	311	часов
Самостоятельная работа под руководством преподавателя	10	10	20	часов
Контрольные работы	2	2	4	часов
Подготовка и сдача экзамена/зачета	4	9	13	часов
Общая трудоемкость (включая промежуточную аттестацию)	144	216	360	часов
			10	з.е.

Формы промежуточной аттестация	Семестр	Количество
Зачет	6	
Контрольные работы	6	1
Экзамен	7	
Курсовой проект	7	
Контрольные работы	7	1

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сенченко П.В.
Должность: Проректор по УР
Дата подписания: 22.02.2023
Уникальный программный ключ:
a1119608-cdff-4455-b54e-5235117c185c

Томск

Согласована на портале № 77178

1. Общие положения

1.1. Цели дисциплины

1. является обучение студентов основам объектно-ориентированного проектирования и программирования в современных средах разработки ПО.

1.2. Задачи дисциплины

1. Получение знаний и практических навыков в области проектирования и разработки объектно-ориентированных программ.
2. Иметь представление о предпосылках возникновения ООП и его месте в эволюции парадигм программирования.
3. Знать принципы объектно-ориентрованного проектирования и программирования.
4. Уметь разрабатывать объектно-ориентированные программы на язык С++.

2. Место дисциплины в структуре ОПОП

Блок дисциплин: Б1. Дисциплины (модули).
Часть блока дисциплин: Обязательная часть.
Модуль дисциплин: Модуль направленности (профиля) (major).
Индекс дисциплины: Б1.О.05.02.
Реализуется с применением электронного обучения, дистанционных образовательных технологий.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Процесс изучения дисциплины направлен на формирование следующих компетенций в соответствии с ФГОС ВО и основной образовательной программой (таблица 3.1):

Таблица 3.1 – Компетенции и индикаторы их достижения

Компетенция	Индикаторы достижения компетенции	Планируемые результаты обучения по дисциплине
Универсальные компетенции		
-	-	-
Общепрофессиональные компетенции		
ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ОПК-4.1. Знает основные стандарты оформления нормативной и технической документации на различных стадиях жизненного цикла информационной системы	Из теории понимает основные стандарты оформления нормативной и технической документации на различных стадиях жизненного цикла информационной системы
	ОПК-4.2. Умеет применять стандарты оформления нормативной и технической документации на различных стадиях жизненного цикла информационной системы	Из практического опыта может применять стандарты оформления нормативной и технической документации на различных стадиях жизненного цикла информационной системы
	ОПК-4.3. Владеет навыками составления нормативной и технической документации на различных этапах жизненного цикла информационной системы	Из теории и практики может пользоваться навыками составления нормативной и технической документации на различных этапах жизненного цикла информационной системы

Согласована на портале № 77178

ОПК-8. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8.1. Знает алгоритмические языки программирования, состав и структуру операционных систем, современные среды разработки программного обеспечения	Из теории понимает алгоритмические языки программирования, состав и структуру операционных систем, современные среды разработки программного обеспечения
	ОПК-8.2. Умеет составлять алгоритмы, разрабатывать программы на алгоритмических языках программирования, тестировать работоспособность программы, интегрировать программные модули	Из практики может составлять алгоритмы, разрабатывать программы на алгоритмических языках программирования, тестировать работоспособность программы, интегрировать программные модули
	ОПК-8.3. Владеет алгоритмическими языками программирования, навыками отладки и тестирования работоспособности программы	Из теории и практики может пользоваться алгоритмическими языками программирования, навыками отладки и тестирования работоспособности программы
Профессиональные компетенции		
-	-	-

4. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 10 зачетных единиц, 360 академических часов.

Распределение трудоемкости дисциплины по видам учебной деятельности представлено в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины по видам учебной деятельности

Виды учебной деятельности	Всего часов	Семестры	
		6 семестр	7 семестр
Контактная аудиторная работа обучающихся с преподавателем, всего	36	16	20
Лабораторные занятия	8	4	4
Курсовой проект	4		4
Самостоятельная работа под руководством преподавателя	20	10	10
Контрольные работы	4	2	2
Самостоятельная работа обучающихся, в т.ч. контактная внеаудиторная работа обучающихся с преподавателем, всего	311	124	187
Самостоятельное изучение тем (вопросов) теоретической части дисциплины	100	38	62
Подготовка к лабораторной работе	71	29	42
Написание отчета по лабораторной работе	72	30	42
Подготовка к контрольной работе	68	27	41
Подготовка и сдача зачета	4	4	
Подготовка и сдача экзамена	9		9

Согласована на портале № 77178

Общая трудоемкость (в часах)	360	144	216
Общая трудоемкость (в з.е.)	10	4	6

5. Структура и содержание дисциплины

5.1. Разделы (темы) дисциплины и виды учебной деятельности

Структура дисциплины по разделам (темам) и видам учебной деятельности приведена в таблице 5.1.

Таблица 5.1 – Разделы (темы) дисциплины и виды учебной деятельности

Названия разделов (тем) дисциплины	Лаб. раб.	Контр. раб.	Курс. пр.	СРП, ч.	Сам. раб., ч	Всего часов (без промежуточной аттестации)	Формируемые компетенции
6 семестр							
1 ВОЗМОЖНОСТИ СИ++	4	2	-	3	51	60	ОПК-4, ОПК-8
2 ОБЪЕКТЫ И КЛАССЫ	-		-	3	58	61	ОПК-4, ОПК-8
3 КОНТЕЙНЕРНЫЕ КЛАССЫ	-		-	4	15	19	ОПК-4, ОПК-8
Итого за семестр	4	2	0	10	124	140	
7 семестр							
4 ПРОИЗВОДНЫЕ КЛАССЫ	4	2	4	4	81	95	ОПК-4, ОПК-8
5 ВИРТУАЛЬНЫЕ ФУНКЦИИ	-			4	81	85	ОПК-4, ОПК-8
6 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ	-			2	25	27	ОПК-4, ОПК-8
Итого за семестр	4	2	4	10	187	207	
Итого	8	4	4	20	311	347	

5.2. Содержание разделов (тем) дисциплины

Содержание разделов (тем) дисциплины приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов (тем) дисциплины

Названия разделов (тем) дисциплины	Содержание разделов (тем) дисциплины	СРП, ч	Формируемые компетенции
6 семестр			
1 ВОЗМОЖНОСТИ СИ++	Локальные и глобальные переменные, подпрограммы и их аргументы, определение данных, операторы динамического распределения памяти, перегрузка функций и операций	3	ОПК-4, ОПК-8
	Итого	3	
2 ОБЪЕКТЫ И КЛАССЫ	Класс как обобщение структуры, определение первичного класса, перегрузка операций, конструкторы, список инициализации, деструктор, дружественные классы, статические элементы класса, шаблоны функций	3	ОПК-4, ОПК-8
	Итого	3	

Согласована на портале № 77178

3 КОНТЕЙНЕРНЫЕ КЛАССЫ	шаблоны классов, параметризованные очереди и стеки, бинарные деревья, определение класса множества	4	ОПК-4, ОПК-8
	Итого	4	
Итого за семестр		10	
7 семестр			
4 ПРОИЗВОДНЫЕ КЛАССЫ	Определение производного класса, доступ к полям и функциям базового, класса, класс дерева поиска, параметризованный связный список, множественное наследование, виртуальные классы	4	ОПК-4, ОПК-8
	Итого	4	
5 ВИРТУАЛЬНЫЕ ФУНКЦИИ	Переопределение составной функции, организация списка объектов, различного типа, техническая реализация виртуальных функций, виртуальные деструкторы, абстрактные классы	4	ОПК-4, ОПК-8
	Итого	4	
6 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ	парадигмы программирования, от процедурного программирования к объектному, основные принципы и этапы объектно-ориентированного программирования, объектно-ориентированные языки программирования, этапы разработки программных систем с использованием ООП, объектная декомпозиция, объекты и сообщения, классы, основные средства разработки классов	2	ОПК-4, ОПК-8
	Итого	2	
Итого за семестр		10	
Итого		20	

5.3. Контрольные работы

Виды контрольных работ и часы на контрольные работы приведены в таблице 5.3.
Таблица 5.3 – Контрольные работы

№ п.п.	Виды контрольных работ	Трудоемкость, ч	Формируемые компетенции
6 семестр			
1	Контрольная работа с автоматизированной проверкой	2	ОПК-4, ОПК-8
Итого за семестр		2	
7 семестр			
2	Контрольная работа с автоматизированной проверкой	2	ОПК-4, ОПК-8
Итого за семестр		2	
Итого		4	

5.4. Лабораторные занятия

Наименование лабораторных работ приведено в таблице 5.4.
Таблица 5.4 – Наименование лабораторных работ

Согласована на портале № 77178

Названия разделов (тем) дисциплины	Наименование лабораторных работ	Трудоемкость, ч	Формируемые компетенции
6 семестр			
1 ВОЗМОЖНОСТИ СИ++	Классы, конструкторы, деструкторы	4	ОПК-4, ОПК-8
	Итого	4	
Итого за семестр		4	
7 семестр			
4 ПРОИЗВОДНЫЕ КЛАССЫ	Перегрузка операторов	4	ОПК-4, ОПК-8
	Итого	4	
Итого за семестр		4	
Итого		8	

5.5.Контроль самостоятельной работы (курсовой проект)

Содержание самостоятельной работы и ее трудоемкость, а также формируемые компетенции в рамках выполнения курсового проекта представлены в таблице 5.5.

Таблица 5.5 – Содержание самостоятельной работы и ее трудоемкость в рамках выполнения курсового проекта

Содержание самостоятельной работы в рамках выполнения курсового проекта	Трудоемкость, ч	Формируемые компетенции
7 семестр		
1. Выбрать тему работы; 2. Разработать структуру программы; 3. Разработать программу на языке программирования С++; 4. Выполнить тестирование программы; 5. Написать отчет.	4	ОПК-4, ОПК-8
Итого за семестр	4	
Итого	4	

Примерная тематика курсовых проектов:

1. Стек.
2. Односвязный список.
3. Бинарное дерево.
4. Неориентированный граф.
5. Ориентированный граф.
6. Матрица.
7. Составное арифметическое выражение.
8. Секундомер.
9. Система линейных алгебраических уравнений.
10. График функции.

5.6. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 5.6.

Таблица 5.6. – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов (тем) дисциплины	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
6 семестр				

Согласована на портале № 77178

1 ВОЗМОЖНОСТИ СИ++	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	12	ОПК-4, ОПК-8	Зачёт, Тестирование
	Подготовка к лабораторной работе	14	ОПК-4, ОПК-8	Лабораторная работа
	Написание отчета по лабораторной работе	15	ОПК-4, ОПК-8	Отчет по лабораторной работе
	Подготовка к контрольной работе	10	ОПК-4, ОПК-8	Контрольная работа
	Итого	51		
2 ОБЪЕКТЫ И КЛАССЫ	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	16	ОПК-4, ОПК-8	Зачёт, Тестирование
	Подготовка к лабораторной работе	15	ОПК-4, ОПК-8	Лабораторная работа
	Написание отчета по лабораторной работе	15	ОПК-4, ОПК-8	Отчет по лабораторной работе
	Подготовка к контрольной работе	12	ОПК-4, ОПК-8	Контрольная работа
	Итого	58		
3 КОНТЕЙНЕРНЫЕ КЛАССЫ	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	10	ОПК-4, ОПК-8	Зачёт, Тестирование
	Подготовка к контрольной работе	5	ОПК-4, ОПК-8	Контрольная работа
	Итого	15		
Итого за семестр		124		
	Подготовка и сдача зачета	4		Зачет
7 семестр				

Согласована на портале № 77178

4 ПРОИЗВОДНЫЕ КЛАССЫ	Подготовка к лабораторной работе	21	ОПК-4, ОПК-8	Лабораторная работа
	Написание отчета по лабораторной работе	21	ОПК-4, ОПК-8	Отчет по лабораторной работе
	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	21	ОПК-4, ОПК-8	Тестирование, Экзамен
	Подготовка к контрольной работе	18	ОПК-4, ОПК-8	Контрольная работа
	Итого	81		
5 ВИРТУАЛЬНЫЕ ФУНКЦИИ	Подготовка к лабораторной работе	21	ОПК-4, ОПК-8	Лабораторная работа
	Написание отчета по лабораторной работе	21	ОПК-4, ОПК-8	Отчет по лабораторной работе
	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	21	ОПК-4, ОПК-8	Тестирование, Экзамен
	Подготовка к контрольной работе	18	ОПК-4, ОПК-8	Контрольная работа
	Итого	81		
6 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ	Самостоятельное изучение тем (вопросов) теоретической части дисциплины	20	ОПК-4, ОПК-8	Тестирование, Экзамен
	Подготовка к контрольной работе	5	ОПК-4, ОПК-8	Контрольная работа
	Итого	25		
Итого за семестр		187		
	Подготовка и сдача экзамена	9		Экзамен
Итого		324		

5.7. Соответствие компетенций, формируемых при изучении дисциплины, и видов учебной деятельности

Соответствие компетенций, формируемых при изучении дисциплины, и видов учебной

Согласована на портале № 77178

деятельности представлено в таблице 5.7.

Таблица 5.7 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Формируемые компетенции	Виды учебной деятельности					Формы контроля
	Лаб. раб.	Курс. пр.	Конт.Раб.	СРП	Сам. раб.	
ОПК-4	+	+	+	+	+	Зачёт, Контрольная работа, Курсовой проект, Лабораторная работа, Отчет по курсовому проекту, Отчет по лабораторной работе, Тестирование, Экзамен
ОПК-8	+	+	+	+	+	Зачёт, Контрольная работа, Курсовой проект, Лабораторная работа, Отчет по курсовому проекту, Отчет по лабораторной работе, Тестирование, Экзамен

6. Рейтинговая система для оценки успеваемости обучающихся

Рейтинговая система не используется

7. Учебно-методическое и информационное обеспечение дисциплины

7.1. Основная литература

1. Катаев М. Ю. Объектно-ориентированное программирование: Учебное пособие / Катаев М. Ю. - Томск: Эль Контент, 2013. - 212 с. Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library.

7.2. Дополнительная литература

1. Морозова Ю. В. Объектно-ориентированный анализ и программирование: Учебное пособие / Морозова Ю. В. - Томск : Эль Контент, 2018. – 140 с. Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library.

7.3. Учебно-методические пособия

7.3.1. Обязательные учебно-методические пособия

1. Катаев М. Ю. Объектно-ориентированное программирование.: Лабораторный практикум / Катаев М. Ю. - Томск: ФДО, ТУСУР, 2014. - 53 с. Доступ из личного кабинета студента. [Электронный ресурс]: — Режим доступа: https://study.tusur.ru/study/library.

7.3.2. Учебно-методические пособия для лиц с ограниченными возможностями здоровья и инвалидов

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа лиц с ограниченными возможностями здоровья и инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Для лиц с нарушениями зрения:

– в форме электронного документа;

– в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

– в форме электронного документа;

– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

– в форме электронного документа;

– в печатной форме.

7.4. Иное учебно-методическое обеспечение

Согласована на портале № 77178

1. Катаев М. Ю. Объектно-ориентированное программирование [Электронный ресурс]: электронный курс / Катаев М. Ю. - Томск: ФДО, ТУСУР, 2018. (доступ из личного кабинета студента) .

7.5. Современные профессиональные базы данных и информационные справочные системы

1. При изучении дисциплины рекомендуется обращаться к современным базам данных, информационно-справочным и поисковым системам, к которым у ТУСУРа открыт доступ: https://lib.tusur.ru/ru/resursy/bazy-dannyh.

8. Материально-техническое и программное обеспечение дисциплины

8.1. Общие требования к материально-техническому и программному обеспечению дисциплины

Учебные аудитории для проведения занятий лабораторного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, для самостоятельной работы студентов

634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- Веб-камера - 6 шт.;
- Наушники с микрофоном - 6 шт.;
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

- 7-Zip;
- Google Chrome;
- Kaspersky Endpoint Security для Windows;
- LibreOffice;
- Microsoft Windows;

8.2. Материально-техническое и программное обеспечение для самостоятельной работы

Для самостоятельной работы используются учебные аудитории (компьютерные классы), расположенные по адресам:

- 634050, Томская область, г. Томск, Ленина проспект, д. 40, 233 ауд.;
- 634045, Томская область, г. Томск, ул. Красноармейская, д. 146, 209 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- учебная мебель;
- компьютеры;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду ТУСУРа.

Перечень программного обеспечения:

- Microsoft Windows;
- OpenOffice;
- Kaspersky Endpoint Security 10 для Windows;
- 7-Zip;
- Google Chrome.

8.3. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины лицами с ограниченными возможностями здоровья и инвалидами осуществляется с использованием средств обучения общего и специального назначения.

При занятиях с обучающимися с **нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах, мобильной системы преподавания

Согласована на портале № 77178

для обучающихся с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой занимаются обучающиеся с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При занятиях с обучающимися с **нарушениями зрения** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для комфортного просмотра.

При занятиях с обучающимися с **нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах, мобильной системы обучения для людей с инвалидностью.

9. Оценочные материалы и методические рекомендации по организации изучения дисциплины

9.1. Содержание оценочных материалов для текущего контроля и промежуточной аттестации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы, представленные в таблице 9.1.

Таблица 9.1 – Формы контроля и оценочные материалы

Названия разделов (тем) дисциплины	Формируемые компетенции	Формы контроля	Оценочные материалы (ОМ)
1 ВОЗМОЖНОСТИ СИ++	ОПК-4, ОПК-8	Зачёт	Перечень вопросов для зачета
		Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Лабораторная работа	Темы лабораторных работ
		Тестирование	Примерный перечень тестовых заданий
		Отчет по лабораторной работе	Темы лабораторных работ
2 ОБЪЕКТЫ И КЛАССЫ	ОПК-4, ОПК-8	Зачёт	Перечень вопросов для зачета
		Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Лабораторная работа	Темы лабораторных работ
		Тестирование	Примерный перечень тестовых заданий
		Отчет по лабораторной работе	Темы лабораторных работ

Согласована на портале № 77178

3 КОНТЕЙНЕРНЫЕ КЛАССЫ	ОПК-4, ОПК-8	Зачёт	Перечень вопросов для зачета
		Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
4 ПРОИЗВОДНЫЕ КЛАССЫ	ОПК-4, ОПК-8	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Лабораторная работа	Темы лабораторных работ
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
		Отчет по лабораторной работе	Темы лабораторных работ
5 ВИРТУАЛЬНЫЕ ФУНКЦИИ	ОПК-4, ОПК-8	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Лабораторная работа	Темы лабораторных работ
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов
		Отчет по лабораторной работе	Темы лабораторных работ
6 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ	ОПК-4, ОПК-8	Контрольная работа	Примерный перечень тем и тестовых заданий на контрольные работы
		Тестирование	Примерный перечень тестовых заданий
		Экзамен	Перечень экзаменационных вопросов

Шкала оценки сформированности отдельных планируемых результатов обучения по дисциплине приведена в таблице 9.2.

Таблица 9.2 – Шкала оценки сформированности планируемых результатов обучения по дисциплине

Оценка	Баллы за ОМ	Формулировка требований к степени сформированности планируемых результатов обучения		
		знать	уметь	владеть

Согласована на портале № 77178

2 (неудовлетворительно)	< 60% от максимальной суммы баллов	отсутствие знаний или фрагментарные знания	отсутствие умений или частично освоенное умение	отсутствие навыков или фрагментарные применение навыков
3 (удовлетворительно)	от 60% до 69% от максимальной суммы баллов	общие, но не структурированные знания	в целом успешно, но не систематически осуществляемое умение	в целом успешное, но не систематическое применение навыков
4 (хорошо)	от 70% до 89% от максимальной суммы баллов	сформированные, но содержащие отдельные проблемы знания	в целом успешное, но содержащие отдельные пробелы умение	в целом успешное, но содержащие отдельные пробелы применение навыков
5 (отлично)	≥ 90% от максимальной суммы баллов	сформированные систематические знания	сформированное умение	успешное и систематическое применение навыков

Шкала комплексной оценки сформированности компетенций приведена в таблице 9.3.

Таблица 9.3 – Шкала комплексной оценки сформированности компетенций

Оценка	Формулировка требований к степени компетенции
2 (неудовлетворительно)	Не имеет необходимых представлений о проверяемом материале или Знать на уровне **ориентирования**, представлений. Обучающийся знает основные признаки или термины изучаемого элемента содержания, их отнесенность к определенной науке, отрасли или объектам, узнает в текстах, изображениях или схемах и знает, к каким источникам нужно обращаться для более детального его усвоения.
3 (удовлетворительно)	Знать и уметь на **репродуктивном** уровне. Обучающихся знает изученный элемент содержания репродуктивно: произвольно воспроизводит свои знания устно, письменно или в демонстрируемых действиях.
4 (хорошо)	Знать, уметь, владеть на **аналитическом** уровне. Зная на репродуктивном уровне, указывать на особенности и взаимосвязи изученных объектов, на их достоинства, ограничения, историю и перспективы развития и особенности для разных объектов усвоения.
5 (отлично)	Знать, уметь, владеть на **системном** уровне. Обучающийся знает изученный элемент содержания системно, произвольно и доказательно воспроизводит свои знания устно, письменно или в демонстрируемых действиях, учитывая и указывая связи и зависимости между этим элементом и другими элементами содержания дисциплины, его значимость в содержании дисциплины.

9.1.1. Примерный перечень тестовых заданий

1. Что такое параметр функции? А. Параметр — это объявление типа данных значения, передаваемого в функцию.
   Б. Параметр — это передаваемый в функцию фактический идентификатор. С. Параметр

Согласована на портале № 77178

— это результат внешней операции, передаваемый в функцию.

2. В чем состоит особенность функции main()? А. Обычная функция, без всяких особенностей Б. Ее вызывает операционная система, начиная выполнение программы. С. Функция main() организует доступ к аппаратному уровню через использование указателей.
3. Какие два типа комментариев вы знаете? А. а) /*комментарий и б) //комментарий// Б. а) /комментарий/ и б) //комментарий// С. а) /*комментарий*/ и б) //комментарий
4. Могут ли комментарии быть вложенными? А. Многострочные комментарии не могут быть вложенными Б. Многострочные комментарии могут быть вложенными С. Многострочные комментарии могут быть вложенными только в функции main()
5. Какой тип комментария является многострочным? А. Комментарий типа /*комментарий*/ называется многострочным, а // однострочным Б. Комментарий типа //комментарий называется многострочным, а /*однострочным*/ С. Комментарий типа //комментарий// называется многострочным, а /*однострочным
6. Могут ли комментарии занимать несколько строк? А. Комментарии не могут занимать несколько строк Б. Да, только тип /*комментарий*/ С. Только занимать несколько строк может тип //комментарий
7. Какую роль выполняет директива #include? А. Директива служит для определения файла, имя которого стоит после директивы, в который предстоит записывать результаты работы программы. Б. Директива служит для определения файла, имя которого стоит после директивы, из которого предстоит считывать данные для работы программы.
   С. Директива служит для введения содержимого файла, имя которого стоит после директивы, в исходный текст программы.
8. Однострочный комментарий может находиться внутри многострочного комментария? А. Да Б. Нет С. Да - только в функции main()
9. Что такое переменные и константы? А. Переменная и константа – это идентификаторы, необходимые для обозначения элементов математических формул, записанных в терминах языка программирования. Б. Переменная – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. Константа – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы. С. Константа – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. переменная – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы.
10. Как выводить значения переменных на экран? А. cin Б. printf С. cout
11. Как вводить значения переменных с экрана? А. cin Б. printf С. cout
12. В каком служебном файле находится описание объектов ввода-вывода? А. Описание объектов для управления вводом-выводом содержится в файле stdafx.h Б. Описание объектов для управления вводом-выводом содержится в файле iostream.h. С. Описание объектов для управления вводом-выводом содержится в файле string.h
13. Есть ли изначально зарезервированные в языке С++ имена переменных? А. Да, только операторы языка Б. Нет С. Да
14. Объявите константу для числа pi, равного 3.14159. А. const float pi = 3.14159 Б. const = 3.14159 С. const pi = 3.14159
15. Объявите переменную типа float и инициализируйте ее, используя константу pi. А. const float pi = 3.14159; myPi = PI; Б. const float PI = 3.14159; float myPi = PI; С. const PI = 3.14159; float myPi = PI;
16. Что такое программа?
    А. Программа представляет собой набор функций, выполняемых в определенной пользователем последовательности.
    Б. Программа представляет собой набор команд и функций, выполняемых в определенной временной последовательности. С. Программа представляет собой набор команд, выполняемых в определенной последовательности.
17. Все выражения в С++ оканчиваются точкой с запятой? А. Да Б. Нет С. Да, есть

Согласована на портале № 77178

исключения

18. Что возвращает выражение? А. В С++ выражения возвращают результаты вычислений. Б. В С++ выражения возвращают результаты вычислений или ничего не делают (нулевые выражения). С. В С++ выражения ничего не делают (нулевые выражения), обозначая те или иные операции. Что такое блок программы? А. Взаимосвязанные по времени выполнения выражения программы объединяют в блоки. Блок начинается открывающей прямой скобкой ([) и оканчивается закрывающей прямой скобкой (]). Б. Логически взаимосвязанные выражения программы объединяют в блоки. Блок начинается открывающей скобкой «(» и оканчивается закрывающей скобкой «)». С0. Логически взаимосвязанные выражения программы объединяют в блоки. Блок начинается открывающей фигурной скобкой ({) и оканчивается закрывающей фигурной скобкой (}).
19. Сколько видов операторов есть в С++? А0. В С++ есть два вида операторов: операторы присваивания и математические операторы. Б. В С++ есть три вида операторов: операторы присваивания, логические и математические операторы. С. В С++ есть четыре вида операторов: операторы присваивания, логические, символьные и математические операторы.
20. Как работает оператор присваивания (=)? А. Оператор присваивания (=) позволяет заменить значение операнда, расположенного с правой стороны от знака равенства, значением, вычисляемым с левой стороны от него. Б. Оператор присваивания (=) позволяет заменить значение операнда, расположенного с левой стороны от знака равенства, значением, вычисляемым с правой стороны от него. С. Оператор присваивания (=) позволяет заменить значение операнда, расположенного с любой стороны, в зависимости от знака.
21. Сколько в С++ используется математических операторов? А. В С++ используется четыре математических оператора: сложения (+), вычитания (-), умножения (*), целочисленного деления (/). Б. В С++ используется шесть математических операторов: сложения (+), вычитания (-), умножения (*), целочисленного деления (/), инвертирования (~) и деления по модулю (%). С. В С++ используется пять математических операторов: сложения (+), вычитания (-), умножения (*), целочисленного деления (/) и деления по модулю (%).
22. Что такое функция? А. Функция — это подпрограмма, которая может манипулировать данными и возвращать некоторое значение. Б. Функция — это подпрограмма, которая может принимать данные, манипулировать ими и возвращать определенное количество значений. С. Функция — это подпрограмма, которая может принимать данные, манипулировать ими и возвращать некоторые значения или ничего не возвращать.
23. Как происходит вызов функции? А. Каждая функция записана в список и вызов автоматически происходит при завершении предыдущей функции из списка и управление переходит к телу последующей функции. Б. Каждая функция обладает собственным именем, и, когда оно встречается в программе, управление переходит к телу данной функции. С. Вызов функции происходит автоматически, после окончания времени выполнения предыдущей функции.
24. В чем отличие языка Си++ от Си? А. Язык Си++ отличается от языка программирования Си, прежде всего, тем, что он поддерживает объектно-ориентированное программирование. Б. Язык Си++ не отличается от языка программирования Си, являясь его расширением. С. Язык Си++ отличается от языка программирования Си, прежде всего, тем, что он поддерживает событийное программирование.

9.1.2. Перечень экзаменационных вопросов

1. Какой из предложенных вариантов записи абстрактного класса в С ++ является правильным: а) class A {virtual int f () = 0;} б) abstract class A {virtual int f () = 0;} в) class A {virtual int f () = 0;} abstract
2. Выберите верное утверждение о деструкторе класса в С ++: а) Деструктор принимает в качестве параметра адрес того объекта, который нужно уничтожить б) Деструктор не содержит параметров в) Деструктор принимает в качестве параметра указатель this
3. Свойство, при котором объекты содержат описание атрибутов и действий одновременно: а) Наследование б) Полиморфизм в) Инкапсуляция
4. Один из принципов объектно-ориентированного программирования: а) Инкапсуляция б)

Согласована на портале № 77178

Ингаляция в) Инструкция

5. Метод определения объектов, при котором производные объекты наследуют свойства от своих потомков: а) Монорфизм б) Полиморфизм в) Наследование
6. Свойство объектов, при котором действие с одинаковыми именами вызывает различное поведение для различных объектов: а) Полиморфизм б) Передача в) Монорфизм
7. Данные, характеризующие состояние объекта: а) Доли объекта б) Части объекта в) Атрибуты объекта
8. Под объектами понимают: а) Всю абстрактную сущность, заданную набором имен атрибутов и имен методов поведения б) Некоторую абстрактную сущность, заданную набором имен атрибутов и имен методов поведения в) Некоторую видимую сущность, заданную набором имен атрибутов и имен методов поведения
9. Совокупность объектов, характеризующаяся общностью методов и свойств: а) Класс б) Вид в) Род
10. Изменение состояния объекта в ответ на какое-либо действие: а) Значение б) Событие в) Данность
11. Действие, которое может выполнить объект: а) Метод б) Событие в) Свойство
12. Характеристика объекта: а) Событие б) Данность в) Свойство
13. Совокупность свойств и методов: а) Объект б) Свойство в) Событие
14. Тип, соответствующий классу: а) Объективный тип б) Объектный тип в) Видимый тип
15. Компоненты, которые видны во время работы приложения, с ними напрямую может взаимодействовать пользователь, называются: а) Абстрактными б) Видимыми в) Визуальными
16. Укажите правильное использование оператора friend: a) class A {public: friend int H :: CountPass (); private: short i;} б) class A {public: friend int H :: q; short i;} в) class A {public: int A1 :: CountPass (); friend: short i;}

9.1.3. Перечень вопросов для зачета

1. Какие утверждения относительно методов-конструкторов являются правильными? 1) конструктор выполняет инициализацию свойств объекта 2) конструктор вызывается раньше всех остальных методов 3) конструктор отвечает за создание объекта при выполнении программы 4) конструктор отвечает за освобождение памяти, выделенной объекту
2. Какие типы свойств может содержать объект? 1) основные стандартные базовые типы 2) структурированные свойства-массивы 3) объектные свойства 4) программные свойства
3. Какие типы методов обычно содержат классы? 1) конструкторы 2) методы доступа к свойствам 3) методы, реализующие функциональность объектов класса 4) деструкторы
4. Какие утверждения относительно понятия «Программный объект» являются правильными? 1) с каждым объектом связываются данные и программный код 2) программный объект является моделью исходного объекта 3) объект существует при выполнении программы, занимая часть оперативной памяти 4) объекты используются для описания только физических сущностей
5. Какие утверждения относительно методов-деструкторов являются правильными? 1) деструктор отвечает за освобождение памяти, выделенной объекту 2) деструкторы реализованы не во всех объектных языках 3) в языках Java и C# вместо деструкторов используется механизм сборки мусора 4) деструктор отсутствуют в языке C++
6. Что определяет принцип инкапсуляции? 1) объединение в рамках объекта некоторых данных и программного кода обработки этих данных 2) ограничение доступа к элементам класса 3) механизм взаимодействия объектов 4) правила обработки ошибочных ситуаций
7. Какие утверждения относительно методов доступа к свойствам являются правильными? 1) директиву class 2) Директиву abstract 3) имя класса 4) имя одного или нескольких интерфейсов
8. Какие компоненты может содержать заголовок абстрактного класса? 1) использование методов доступа является одним из проявлений принципа инкапсуляции 2) для изменения значения свойства используются Set-методы (сеттеры) 3) для запроса значения свойства вводятся Get-методы (геттеры) 4) методы доступа всегда являются абстрактными
9. Какие утверждения относительно абстрактных методов являются правильными? 1) Абстрактный метод имеет заголовок, но не имеет программной реализации 2) класс

Согласована на портале № 77178

может содержать несколько абстрактных методов 3) абстрактные методы объявляются с помощью директивы abstract 4) все абстрактные методы в классе обязательно должны иметь одно и то же имя

10. Какие уровни доступа могут иметь элементы классов? 1) открытый (public) 2) защищенный (protected) 3) закрытый (private) 4) привилегированный (privileged)
11. Какие утверждения справедливы относительно методов объекта? 1) набор методов, связанных с объектом, не изменяется при выполнении программы 2) методы могут быть открытыми или закрытыми 3) методы определяют поведение объекта 4) метод всегда содержит некоторый программный код
12. Какие утверждения относительно открытых элементов класса являются правильными? 1) открытые элементы доступны в любом месте программы 2) открытые элементы задаются директивой public 3) открытыми элементами обычно являются методы класса 4) открытыми элементами всегда являются свойства класса
13. Какие утверждения справедливы относительно правил описания методов? 1) методы оформляются как подпрограммы 2) в одном классе методы могут иметь одинаковые имена 3) методы могут быть объявлены абстрактными 4) методы могут быть объявлены как статические, классовые
14. Как синтаксически правильно оформляется вызов метода для объектной переменной? 1) ИмяПеременной.ИмяМетода; 2) ИмяМетода.ИмяПеременной; 3) ИмяМетода(ИмяПеременной); 4) ИмяПеременной:ИмяМетода;
15. Какие утверждения справедливы для операции присваивания значений объектных переменных? 1) в простейшем случае присваивать можно значения только переменных одного и того же класса 2) после выполнения операции обе переменные будут адресовать один и тот же объект 3) любой объектной переменной можно присвоить пустое значение 4) операция присваивания приводит к созданию копии адресуемого объекта
16. Какие утверждения справедливы относительно понятия «объектная переменная»? 1) объектная переменная используется для доступа к свойствам и методам объекта 2) объектная переменная должна иметь имя 3) при объявлении объектной переменной обязательно указывается класс, с которым она связывается 4) объектная переменная является скрытым, неявным указателем (ссылкой) на размещаемый в памяти объект
17. Какие утверждения справедливы относительно процесса создания объектов? 1) при успешном создании объекта устанавливается адресное значение объектной переменной 2) для создания объекта вызывается метод-конструктор 3) создание объектов происходит при объявлении объектных переменных 4) создание объектов синтаксически оформляется как оператор присваивания
18. Какие утверждения справедливы для операции сравнения значений объектных переменных? 1) сравнивать можно только переменные одно и того же классового типа 2) операция выполняет сравнение адресных значений объектных переменных 3) любую объектную переменную можно сравнивать с пустым значением 4) операция сравнения выполняет сравнение значений свойств соответствующих объектов
19. Какие конструкции реализуют правильное создание объектов класса MyClass в языках Java и C# ? 1) MyClass MyObject; MyObject = new MyClass( ); 2) MyClass MyObject = new MyClass( ); 3) MyClass MyObject; new(MyObject) ; 4) MyClass MyObject;
20. Какие операции определены для объектных переменных? 1) присваивание значения одной переменной другой 2) сравнение значений двух переменных на равенство или неравенство 3) сравнение значений двух переменных на больше-меньше 4) любые арифметические операции

9.1.4. Примерный перечень тематик курсовых проектов

1. Стек.
2. Односвязный список.
3. Бинарное дерево.
4. Неориентированный граф.
5. Ориентированный граф.
6. Матрица.
7. Составное арифметическое выражение.
8. Секундомер.

Согласована на портале № 77178

9. Система линейных алгебраических уравнений.
10. График функции.

9.1.5. Примерный перечень тем и тестовых заданий на контрольные работы

1. Что такое параметр функции?
   А. Параметр — это объявление типа данных значения, передаваемого в функцию.
   Б. Параметр — это передаваемый в функцию фактический идентификатор.
   С. Параметр — это результат внешней операции, передаваемый в функцию.
2. В чем состоит особенность функции main()?
   А. Обычная функция, без всяких особенностей
   Б. Ее вызывает операционная система, начиная выполнение программы.
   С. Функция main() организует доступ к аппаратному уровню через использование указателей.
3. Какие два типа комментариев вы знаете?
   А. а) /*комментарий и б) //комментарий//
   Б. а) /комментарий/ и б) //комментарий//
   С. а) /*комментарий*/ и б) //комментарий
4. Могут ли комментарии быть вложенными?
   А. Многострочные комментарии не могут быть вложенными
   Б. Многострочные комментарии могут быть вложенными
   С. Многострочные комментарии могут быть вложенными только в функции main()
5. Какой тип комментария является многострочным?
   А. Комментарий типа /*комментарий*/ называется многострочным, а // однострочным
   Б. Комментарий типа //комментарий называется многострочным, а /*однострочным*/
   С. Комментарий типа //комментарий// называется многострочным, а /*однострочным
6. Могут ли комментарии занимать несколько строк?
   А. Комментарии не могут занимать несколько строк
   Б. Да, только тип /*комментарий*/
   С. Только занимать несколько строк может тип //комментарий
7. Какую роль выполняет директива #include?
   А. Директива служит для определения файла, имя которого стоит после директивы, в который предстоит записывать результаты работы программы.
   Б. Директива служит для определения файла, имя которого стоит после директивы, из которого предстоит считывать данные для работы программы.
   С. Директива служит для введения содержимого файла, имя которого стоит после директивы, в исходный текст программы.
8. Однострочный комментарий может находиться внутри многострочного комментария?
   А. Да
   Б. Нет
   С. Да - только в функции main()
9. Что такое переменные и константы?
   А. Переменная и константа – это идентификаторы, необходимые для обозначения элементов математических формул, записанных в терминах языка программирования.
   Б0. Переменная – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. Константа – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы.
   С. Константа – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. переменная – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы.
10. Как выводить значения переменных на экран?
    А. cin
    Б. printf
    С. cout

Согласована на портале № 77178

11. Могут ли комментарии занимать несколько строк?
    А. Комментарии не могут занимать несколько строк
    Б. Да, только тип /*комментарий*/
    С. Только занимать несколько строк может тип //комментарий
12. Какую роль выполняет директива #include?
    А. Директива служит для определения файла, имя которого стоит после директивы, в который предстоит записывать результаты работы программы.
    Б. Директива служит для определения файла, имя которого стоит после директивы, из которого предстоит считывать данные для работы программы.
    С. Директива служит для введения содержимого файла, имя которого стоит после директивы, в исходный текст программы.
13. Однострочный комментарий может находиться внутри многострочного комментария?
    А. Да
    Б. Нет
    С. Да - только в функции main()
14. Что такое переменные и константы?
    А. Переменная и константа – это идентификаторы, необходимые для обозначения элементов математических формул, записанных в терминах языка программирования.
    Б. Переменная – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. Константа – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы.
    С. Константа – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. переменная – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы.
15. Как выводить значения переменных на экран?
    А. cin
    Б. printf
    С. cout
16. Как вводить значения переменных с экрана?
    А. cin
    Б. printf
    С. cout
17. Что такое параметр функции?
    А. Параметр — это объявление типа данных значения, передаваемого в функцию.
    Б. Параметр — это передаваемый в функцию фактический идентификатор.
    С. Параметр — это результат внешней операции, передаваемый в функцию.
18. В чем состоит особенность функции main()?
    А. Обычная функция, без всяких особенностей
    Б. Ее вызывает операционная система, начиная выполнение программы.
    С. Функция main() организует доступ к аппаратному уровню через использование указателей.
19. Могут ли комментарии быть вложенными?
    А. Многострочные комментарии не могут быть вложенными
    Б. Многострочные комментарии могут быть вложенными
    С. Многострочные комментарии могут быть вложенными только в функции main()
20. Какую роль выполняет директива #include?
    А. Директива служит для определения файла, имя которого стоит после директивы, в который предстоит записывать результаты работы программы.
    Б. Директива служит для определения файла, имя которого стоит после директивы, из которого предстоит считывать данные для работы программы.
    С. Директива служит для введения содержимого файла, имя которого стоит после директивы, в исходный текст программы.
21. Что такое переменные и константы?

Согласована на портале № 77178

А. Переменная и константа – это идентификаторы, необходимые для обозначения элементов математических формул, записанных в терминах языка программирования.
Б. Переменная – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. Константа – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы.
С. Константа – это ячейка для хранения информации, из которой имеется возможность неоднократно считывать значение переменной, а также возможность записывать в эту ячейку другое значение. Переменная может изменять свое значение в процессе выполнения программы. переменная – это ячейка, в которой значение может оставаться неизменным от начала до конца выполнения программы.

9.1.6. Темы лабораторных работ

1. Классы, конструкторы, деструкторы
2. Перегрузка операторов

9.2. Методические рекомендации

Учебный материал излагается в форме, предполагающей самостоятельное мышление студентов, самообразование. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

Начать изучение дисциплины необходимо со знакомства с рабочей программой, списком учебно-методического и программного обеспечения. Самостоятельная работа студента включает работу с учебными материалами, выполнение контрольных мероприятий, предусмотренных учебным планом.

В процессе изучения дисциплины для лучшего освоения материала необходимо регулярно обращаться к рекомендуемой литературе и источникам, указанным в учебных материалах; пользоваться через кабинет студента на сайте Университета образовательными ресурсами электронно-библиотечной системы, а также общедоступными интернет-порталами, содержащими научно-популярные и специализированные материалы, посвященные различным аспектам учебной дисциплины.

При самостоятельном изучении тем следуйте рекомендациям:

– чтение или просмотр материала осуществляйте со скоростью, достаточной для индивидуального понимания и освоения материала, выделяя основные идеи; на основании изученного составить тезисы. Освоив материал, попытаться соотнести теорию с примерами из практики;

– если в тексте встречаются незнакомые или малознакомые термины, следует выяснить их значение для понимания дальнейшего материала;

– осмысливайте прочитанное и изученное, отвечайте на предложенные вопросы.

Студенты могут получать индивидуальные консультации, в т.ч. с использованием средств телекоммуникации.

По дисциплине могут проводиться дополнительные занятия, в т.ч. в форме вебинаров. Расписание вебинаров и записи вебинаров публикуются в электронном курсе по дисциплине.

9.3. Требования к оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусмотрены дополнительные оценочные материалы, перечень которых указан в таблице 9.4.

Таблица 9.4 – Дополнительные материалы оценивания для лиц с ограниченными возможностями здоровья и инвалидов

Категории обучающихся	Виды дополнительных оценочных материалов	Формы контроля и оценки результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка

Согласована на портале № 77178

С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами, определяющимися исходя из состояния обучающегося на момент проверки

9.4. Методические рекомендации по оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусматривается доступная форма предоставления заданий оценочных средств, а именно:

– в печатной форме;
– в печатной форме с увеличенным шрифтом;
– в форме электронного документа;
– методом чтения ассистентом задания вслух;
– предоставление задания с использованием сурдоперевода.

Лицам с ограниченными возможностями здоровья и инвалидам увеличивается время на подготовку ответов на контрольные вопросы. Для таких обучающихся предусматривается доступная форма предоставления ответов на задания, а именно:

– письменно на бумаге;
– набор ответов на компьютере;
– набор ответов с использованием услуг ассистента;
– представление ответов устно.

Процедура оценивания результатов обучения лиц с ограниченными возможностями здоровья и инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:
– в форме электронного документа;
– в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:
– в форме электронного документа;
– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:
– в форме электронного документа;
– в печатной форме.

При необходимости для лиц с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 77178

ЛИСТ СОГЛАСОВАНИЯ

Рассмотрена и одобрена на заседании кафедры АСУ
протокол № 1 от «24» 1 2023 г.

СОГЛАСОВАНО:

Должность	Инициалы, фамилия	Подпись
Заведующий выпускающей каф. АСУ	В.В. Романенко	Согласовано, c3e2018f-3231-48c3-b093-89b6f5342191
Заведующий обеспечивающей каф. АСУ	В.В. Романенко	Согласовано, c3e2018f-3231-48c3-b093-89b6f5342191
Декан ФДО	И.П. Черкашина	Согласовано, 4580bdea-d7a1-4d22-bda1-21376d739cfc

ЭКСПЕРТЫ:

Доцент, каф. АСУ	А.И. Исакова	Согласовано, 79bf1038-9d22-4279-a1e8-7806307b7f82
Доцент, каф. АСУ	А.И. Исакова	Согласовано, 79bf1038-9d22-4279-a1e8-7806307b7f82

РАЗРАБОТАНО:

Профессор, каф. АСУ	М.Ю. Катаев	Разработано, 929f34b8-0cef-484f-b3aa-9d71c10f8183

Согласована на портале № 77178