МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

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

УТВЕРЖДАЮ
Директор департамента образования
_____ П. Е. Троян
«___» _____ 20__ г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки / специальность: **38.03.05 Бизнес-информатика**
Направленность (профиль) / специализация: **ИТ-предпринимательство**
Форма обучения: **заочная (в том числе с применением дистанционных образовательных технологий)**
Факультет: **ФДО, Факультет дистанционного обучения**
Кафедра: **АОИ, Кафедра автоматизации обработки информации**
Курс: **2**
Семестр: **4**
Учебный план набора 2014 года

Распределение рабочего времени

№	Виды учебной деятельности	4 семестр	Всего	Единицы
1	Самостоятельная работа под руководством преподавателя	12	12	часов
2	Лабораторные работы	8	8	часов
3	Контроль самостоятельной работы	2	2	часов
4	Всего контактной работы	22	22	часов
5	Самостоятельная работа	113	113	часов
6	Всего (без экзамена)	135	135	часов
7	Подготовка и сдача экзамена	9	9	часов
8	Общая трудоемкость	144	144	часов
			4.0	З.Е.

Контрольные работы: 4 семестр - 1
Экзамен: 4 семестр

Томск 2018

Согласована на портале № 45603

Рассмотрена и одобрена на заседании кафедры
протокол № 318 от « 3 » 7 2018 г.

ЛИСТ СОГЛАСОВАНИЯ

Рабочая программа дисциплины составлена с учетом требований федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 38.03.05 Бизнес-информатика, утвержденного 11.08.2016 года, рассмотрена и одобрена на заседании кафедры АОИ «___» _____ 20__ года, протокол №_____.

Разработчики:

старший преподаватель каф. АОИ _____ Н. В. Пермякова

доцент каф. АОИ _____ Ю. В. Морозова

Заведующий обеспечивающей каф. АОИ _____ Ю. П. Ехлаков

Рабочая программа дисциплины согласована с факультетом и выпускающей кафедрой:

Декан ФДО _____ И. П. Черкашина

Заведующий выпускающей каф. АОИ _____ Ю. П. Ехлаков

Эксперты:

Доцент кафедры технологий электронного обучения (ТЭО) _____ Ю. В. Морозова

Доцент кафедры автоматизации обработки информации (АОИ) _____ А. А. Сидоров

Согласована на портале № 45603

1. Цели и задачи дисциплины

1.1. Цели дисциплины

Сформировать способность использования компьютерной техники для решения прикладных задач.

Сформировать навыки обработки информации с использованием компьютерной техники.

Сформировать способность применения базовых алгоритмов для решения практических профессиональных задач.

Развить практические навыки структурного программирования.

1.2. Задачи дисциплины

– Ознакомить студентов с различными способами представления обрабатываемых данных;

– изучить различные способы сортировки данных;

– рассмотреть понятие эффективности алгоритмов;

– научить применять теоретические знания, полученные при изучении математических дисциплин на практике.

2. Место дисциплины в структуре ОПОП

Дисциплина «Программирование» (Б1.Б.19) относится к блоку 1 (базовая часть).

Предшествующими дисциплинами, формирующими начальные знания, являются: Дискретная математика, Информатика.

Последующими дисциплинами являются: Базы данных, Объектно-ориентированный анализ и программирование.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

– ОПК-3 способностью работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях;

В результате изучения дисциплины обучающийся должен:

– **знать** базовые структуры представления информации в компьютерных программах; алгоритмы сортировки массивов, способы оценки эффективности алгоритмов; машинные способы представления графов; алгоритмы генерации комбинаторных алгоритмов

– **уметь** разрабатывать алгоритмы прикладных задач; выполнять осознанный выбор структуры представления данных в компьютерной программе;

– **владеть** навыками реализации и отладки программ на языке программирования Си.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 4.0 зачетных единицы и представлена в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры
		4 семестр
Контактная работа (всего)	22	22
Самостоятельная работа под руководством преподавателя (СРП)	12	12
Лабораторные работы	8	8
Контроль самостоятельной работы (КСР)	2	2
Самостоятельная работа (всего)	113	113
Подготовка к контрольным работам	20	20
Оформление отчетов по лабораторным работам	6	6
Подготовка к лабораторным работам	6	6

Согласована на портале № 45603

Самостоятельное изучение тем (вопросов) теоретической части курса	81	81
Всего (без экзамена)	135	135
Подготовка и сдача экзамена	9	9
Общая трудоемкость, ч	144	144
Зачетные Единицы	4.0	

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

Названия разделов дисциплины	СРП, ч	Лаб. раб., ч	КСР, ч	Сам. раб., ч	Всего часов (без экзамена)	Формируемые компетенции
4 семестр						
1 Основы алгоритмизации	1	0	2	12	13	ОПК-3
2 Интегрированная среда программирования DEV-CPP	1	0		14	15	ОПК-3
3 Синтаксис и алфавит языка Си	2	8		22	32	ОПК-3
4 Типы данных языка Си	2	0		12	14	ОПК-3
5 Подготовка и исполнение программы на языке Си	1	0		11	12	ОПК-3
6 Конструкции структурного программирования в Си	1	0		10	11	ОПК-3
7 Функции	1	0		8	9	ОПК-3
8 Массивы	1	0		8	9	ОПК-3
9 Файлы в Си	1	0		10	11	ОПК-3
10 Управление выводом в консоль	1	0		6	7	ОПК-3
Итого за семестр	12	8	2	113	135	
Итого	12	8	2	113	135	

5.2. Содержание разделов дисциплины (самостоятельная работа под руководством преподавателя)

Содержание разделов дисциплин (самостоятельная работа под руководством преподавателя) приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов дисциплин (самостоятельная работа под руководством преподавателя)

Названия разделов	Содержание разделов дисциплины (самостоятельная работа под руководством преподавателя)	Трудоемкость, ч	Формируемые компетенции
4 семестр			
1 Основы алгоритмизации	Основные понятия и определения. Типы данных. Структурное программирование. Системы кодирования алгоритмов.	1	ОПК-3

Согласована на портале № 45603

	Система псевдокод. Блок-диаграммы. Диаграммы Насси − Шнейдермана. Основные алгоритмы. Алгоритмы суммы и произведения. Алгоритмы поиска		
	Итого	1	
2 Интегрированная среда программирования DEV-CPP	Подготовка программного обеспечения для работы в среде DEV-CPP. Настройка параметров среды. Создание проекта. Компиляция и выполнение. Отладка программы. Многофайловая компиляция. Сообщения об ошибках	1	ОПК-3
	Итого	1	
3 Синтаксис и алфавит языка Си	Алфавит языка Си. Синтаксис. Лексемы языка. Ключевые слова. Идентификаторы. Константы. Литеральные строки. Операторы. Знаки пунктуации	2	ОПК-3
	Итого	2	
4 Типы данных языка Си	Основные типы данных. Простые типы. Приставки к типам данных. Преобразование типов. Производные типы данных. Указатели. Ссылки. Разыменование указателей. Сложные типы данных. Массивы. Структуры. Объединения. Перечисления. Объявления и инициализация переменных	2	ОПК-3
	Итого	2	
5 Подготовка и исполнение программы на языке Си	Этапы подготовки программы к исполнению. Директивы препроцессора. Директива #include. Директива #include_next. Директивы #define, #undef, #ifdef, #ifndef. Директивы условной компиляции. Управляющая директива #line. Директива #error. Директива #pragma. Ввод-вывод информации. Функция printf. Функция scanf. Простая программа на языке Си	1	ОПК-3
	Итого	1	
6 Конструкции структурного программирования в Си	Следование. Ветвление. Оператор проверки условия if <else>. Множественный выбор. Циклы. Цикл с фиксированным числом операций for. Циклы while и do while. Операторы безусловной передачи управления continue и break. Примеры использования операторов цикла. Вычисление суммы бесконечного ряда. Вычисления по итерационной формуле. Программирование численных методов	1	ОПК-3
	Итого	1	

Согласована на портале № 45603

7 Функции	Синтаксис. Объявление и вызов функций. Локальные переменные. Выход из функций. Передача параметров по ссылке. Рекурсивные функции	1	ОПК-3
	Итого	1	
8 Массивы	Одномерные массивы. Инициализация массива. Поиск значений в массиве. Сортировка массивов. Многомерные массивы. Инициализация матриц. Печать матриц. Примеры решений задач с использованием матриц. Строки. Инициализация строк. Представление строки в памяти компьютера. Стандартные функции для работы со строками	1	ОПК-3
	Итого	1	
9 Файлы в Си	Типы файлов в Си. Механизм чтения-записи. Функции для поточного доступа к файлам. Примеры работы с текстовыми файлами. Запись данных в текстовый файл. Чтение данных из текстового файла. Изменение текстового файла. Двоичные файлы. Запись и чтение информации в двоичный файл. Реализация прямого доступа в двоичном файле	1	ОПК-3
	Итого	1	
10 Управление выводом в консоль	Win32 API. Типы данных Windows. Функции WinAPI	1	ОПК-3
	Итого	1	
Итого за семестр		12	

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами

Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами представлены в таблице 5.3.

Таблица 5.3 – Разделы дисциплины и междисциплинарные связи

Наименование дисциплин	№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих и обеспечиваемых дисциплин									
	1	2	3	4	5	6	7	8	9	10
Предшествующие дисциплины										
1 Дискретная математика						+				
2 Информатика	+	+	+	+	+	+	+	+	+	+
Последующие дисциплины										
1 Базы данных	+			+		+				
2 Объектно-ориентированный анализ и программирование	+	+	+	+				+	+	+

Согласована на портале № 45603

5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий представлено в таблице 5.4.

Таблица 5.4 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Компетенции	Виды занятий				Формы контроля
	СРП	Лаб. раб.	КСР	Сам. раб.	
ОПК-3	+	+	+	+	Контрольная работа, Экзамен, Проверка контрольных работ, Отчет по лабораторной работе, Тест

6. Интерактивные методы и формы организации обучения

Не предусмотрено РУП.

7. Лабораторные работы

Наименование лабораторных работ приведено в таблице 7.1.

Таблица 7.1 – Наименование лабораторных работ

Названия разделов	Наименование лабораторных работ	Трудоемкость, ч	Формируемые компетенции
4 семестр			
3 Синтаксис и алфавит языка Си	Проверка условий	4	ОПК-3
	Обработка массивов	4	
	Итого	8	
Итого за семестр		8	

8. Контроль самостоятельной работы

Виды контроля самостоятельной работы приведены в таблице 8.1.

Таблица 8.1 – Виды контроля самостоятельной работы

№	Вид контроля самостоятельной работы	Трудоемкость (час.)	Формируемые компетенции
4 семестр			
1	Контрольная работа с автоматизированной проверкой	2	ОПК-3
Итого		2	

9. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 9.1.

Таблица 9.1 – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
4 семестр				
1 Основы алгоритмизации	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ОПК-3	Контрольная работа, Отчет по лабораторной работе, Тест, Экзамен
	Подготовка к контрольным работам	2		

Согласована на портале № 45603

	Итого	12		
2 Интегрированная среда программирования DEV-CPP	Самостоятельное изучение тем (вопросов) теоретической части курса	12	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	14		
3 Синтаксис и алфавит языка Си	Самостоятельное изучение тем (вопросов) теоретической части курса	8	ОПК-3	Контрольная работа, Отчет по лабораторной работе, Тест, Экзамен
	Подготовка к лабораторным работам	6		
	Оформление отчетов по лабораторным работам	6		
	Подготовка к контрольным работам	2		
	Итого	22		
4 Типы данных языка Си	Самостоятельное изучение тем (вопросов) теоретической части курса	10	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	12		
5 Подготовка и исполнение программы на языке Си	Самостоятельное изучение тем (вопросов) теоретической части курса	9	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	11		
6 Конструкции структурного программирования в Си	Самостоятельное изучение тем (вопросов) теоретической части курса	8	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	10		
7 Функции	Самостоятельное изучение тем (вопросов) теоретической части курса	6	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		

Согласована на портале № 45603

	Итого	8		
8 Массивы	Самостоятельное изучение тем (вопросов) теоретической части курса	6	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	8		
9 Файлы в Си	Самостоятельное изучение тем (вопросов) теоретической части курса	8	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	10		
10 Управление выводом в консоль	Самостоятельное изучение тем (вопросов) теоретической части курса	4	ОПК-3	Контрольная работа, Тест, Экзамен
	Подготовка к контрольным работам	2		
	Итого	6		
	Выполнение контрольной работы	2	ОПК-3	Контрольная работа
Итого за семестр		113		
	Подготовка и сдача экзамена	9		Экзамен
Итого		122		

10. Контроль самостоятельной работы (курсовой проект / курсовая работа)

Не предусмотрено РУП.

11. Рейтинговая система для оценки успеваемости обучающихся

Рейтинговая система не используется.

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основная литература

1. Пермякова Н.В. Информатика и программирование [Электронный ресурс]: учебное пособие.Томск : ФДО, ТУСУР, 2016. Доступ из личного кабинета студентов. — Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 10.09.2018).

12.2. Дополнительная литература

1. Каширин, И.Ю. От Си к Си++ [Электронный ресурс]: учебное пособие / И.Ю. Каширин, В.С. Новичков. — Электрон. дан. — Москва Горячая линия-Телеком, 2012. — 334 с. Доступ из личного кабинета студента. — Режим доступа: https://e.lanbook.com/book/5161 (дата обращения: 10.09.2018).

2. Подбельский, В.В. Курс программирования на языке Си [Электронный ресурс] [Электронный ресурс]: учебник / В.В. Подбельский, С.С. Фомин. — Электрон. дан. — Москва : ДМК Пресс, 2012. — 384 с. — Режим доступа: https://e.lanbook.com/book/4148 (дата обращения: 10.09.2018).

Согласована на портале № 45603

12.3. Учебно-методические пособия

12.3.1. Обязательные учебно-методические пособия

1. Пермякова Н. В. Информатика и программирование : электронный курс / Н.В. Пермякова. – Томск : ФДО ТУСУР, 2016. Доступ из личного кабинета студента.

2. Пермякова Н. В. Информатика и программирование [Электронный ресурс]: методические указания по выполнению лабораторных работ. – Томск : ФДО ТУСУР, 2016. Доступ из личного кабинета студента. — Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 10.09.2018).

3. Пермякова Н.В. Программирование [Электронный ресурс]: методические указания по организации самостоятельной работы для студентов заочной формы обучения направления подготовки 38.03.05 Бизнес-информатика, обучающихся с применением дистанционных образовательных технологий / Н. В. Пермякова. Ю.В. Морозова . – Томск : ФДО, ТУСУР, 2018. Доступ из личного кабинета студента. — Режим доступа: https://study.tusur.ru/study/library/ (дата обращения: 10.09.2018).

12.3.2. 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.

12.4. Профессиональные базы данных и информационные справочные системы

1. Дополнительно к профессиональным базам данных рекомендуется использовать информационные, справочные и нормативные базы данных https://lib.tusur.ru/ru/resursy/bazy-dannyh (со свободным доступом).

2. ЭБС «Лань»: www.e.lanbook.com (доступ из личного кабинета студента по ссылке http://lanbook.fdo.tusur.ru).

13. Материально-техническое обеспечение дисциплины и требуемое программное обеспечение

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

13.1.1. Материально-техническое и программное обеспечение дисциплины

Кабинет для самостоятельной работы студентов

учебная аудитория для проведения занятий лабораторного типа, помещение для проведения групповых и индивидуальных консультаций, помещение для проведения текущего контроля и промежуточной аттестации, помещение для самостоятельной работы

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

Описание имеющегося оборудования:

- Коммутатор MicroTeak;
- Компьютер PENTIUM D 945 (3 шт.);
- Компьютер GELERON D 331 (2 шт.);
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

Согласована на портале № 45603

– 7-zip (с возможностью удаленного доступа)
– DEV C++ (с возможностью удаленного доступа)
– Google Chrome
– Kaspersky Endpoint Security 10 для Windows (с возможностью удаленного доступа)
– MS Office версий 2010 (с возможностью удаленного доступа)
– Microsoft Windows
– Notepad++ (с возможностью удаленного доступа)
– OpenOffice (с возможностью удаленного доступа)

13.1.2. Материально-техническое и программное обеспечение для лабораторных работ

Кабинет для самостоятельной работы студентов

учебная аудитория для проведения занятий лабораторного типа, помещение для проведения групповых и индивидуальных консультаций, помещение для проведения текущего контроля и промежуточной аттестации, помещение для самостоятельной работы

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

Описание имеющегося оборудования:

- Коммутатор MicroTeak;
- Компьютер PENTIUM D 945 (3 шт.);
- Компьютер GELERON D 331 (2 шт.);
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:

– 7-zip (с возможностью удаленного доступа)
– DEV C++ (с возможностью удаленного доступа)
– Google Chrome
– Kaspersky Endpoint Security 10 для Windows (с возможностью удаленного доступа)
– MS Office версий 2010 (с возможностью удаленного доступа)
– Microsoft Windows
– OpenOffice (с возможностью удаленного доступа)

13.1.3. 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, 201 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Состав оборудования:

- учебная мебель;
- компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 5 шт.;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду университета.

Перечень программного обеспечения:

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

13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными

Согласована на портале № 45603

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 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 зрениями** 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.

14. Оценочные материалы и методические рекомендации по организации изучения дисциплины

14.1. Содержание оценочных материалов и методические рекомендации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы в составе:

14.1.1. Тестовые задания

Вопрос 1.

Обрабатываемый в программе набор числовых данных сохранен в однонаправленном динамическом линейном списке, который организован по правилу стека. Текущее состояние списка выглядит следующим образом: Head -> 1 2 3 4 5 6 7 -> NULL. Как будет выглядеть список после добавления элемента со значением 8?

Head-> 1 2 3 4 5 6 7 8 -> NULL

Head -> 8 1 2 3 4 5 6 7 -> NULL

Head ->8 1 2 3 4 5 6 7 8 -> NULL

Head -> 1 8 2 3 4 5 6 7 -> NULL

Вопрос 2.

Обрабатываемый в программе набор числовых данных сохранен в однонаправленном динамическом линейном списке, который организован по правилу очереди. Текущее состояние списка выглядит следующим образом: Head -> 1 2 3 4 5 6 7 -> NULL Как будет выглядеть список после добавления элемента со значением 8?

Head -> 1 2 3 4 5 6 7 8 -> NULL

Неад -> 8 1 2 3 4 5 6 7 -> NULL

Неад ->8 1 2 3 4 5 6 7 8 -> NULL

Неад -> 1 8 2 3 4 5 6 7 -> NULL

Вопрос 3.

Обрабатываемая в программе информация хранится в двоичном файле. Какая функция языка Си передвигает указатель чтения-записи файла в заданную позицию файла?

ftell

gopos

stepgo

fseek

Вопрос 4.

Обрабатываемый в программе набор данных 2 11 14 3 14 18 1 14 4 19 0 17 17 6 1 23 хранится в двоичном файле. Что будет выведено на экран при выполнении следующего фрагмента программы?

```
f = fopen("file1","rb");
for(i=0;i<4;i++){
```

Согласована на портале № 45603

```
fseek(f, i*4*sizeof(int) + i*sizeof(int),SEEK_SET);
fread(&x,sizeof(int),1,f);
printf("%d ",x);
}
```

2 14 4 17
2 11 14 3
2 18 0 23
3 1 19 17

Вопрос 5.

Обрабатываемый в программе набор данных 13 28 24 6 2 25 30 18 18 14 30 31 10 6 30 17 хранится в двоичном файле. Что будет выведено на экран при выполнении следующего фрагмента программы?

```
f = fopen("file1","rb");
for(i=0;i<4;i++){
fseek(f,i*4*sizeof(int) + sizeof(int),SEEK_SET);
fread(&x,sizeof(int),1,f);
printf("%d ",x);
}
```

28 25 14 6
13 28 24 6
18 14 30 31
10 6 30 17

Вопрос 6.

Сортировкой элементов массива будем называть такой способ обработки информации, который изменяет порядок элементов массива по следующему правилу: каждое ai ≤ ai+1 Какой из алгоритмов сортировки сравнивает два рядом стоящих элемента и меняет их местами, если первый элемент пары больше, чем второй?

сортировка вставками
сортировка выбором
сортировка бинарными вставками
сортировка обменом

Вопрос 7.

Сортировкой элементов массива будем называть такой способ обработки информации, который изменяет порядок элементов массива по следующему правилу: каждое ai ≤ ai+1. Какой из алгоритмов сортировки ищет минимальный элемент массива и меняет его местами с первым элементом, затем ищет минимальный элемент среди оставшихся, и меняет его местами со вторым элементом и так далее, пока не будет отсортирована вся последовательность?

сортировка вставками
сортировка выбором
сортировка бинарными вставками
сортировка обменом

Вопрос 8.

Сортировкой элементов массива будем называть такой способ обработки информации, который изменяет порядок элементов массива по следующему правилу: каждое ai ≤ ai+1. Какое из представленных утверждений верно?

сортировка вставками устойчива
сортировка обменом не устойчива
сортировка выбором устойчива
сортировка вставками – неестественная сортировка

Вопрос 9.

Сортировкой элементов массива будем называть такой способ обработки информации, который изменяет порядок элементов массива по следующему правилу: ai ≤ ai+1. Сортировка пирамидой (HeapSort) на первом этапе алгоритма строит пирамидально упорядоченный массив. В исходном массиве элементы располагались следующим образом:

Согласована на портале № 45603

1 6 2 4 1 7 9 3

Каким образом будут располагаться элементы массива после выполнения первого этапа HeapSort?

1 6 2 4 1 7 9 3
1 1 2 3 4 6 7 9
3 2 1 1 4 7 6 9
9 6 7 4 1 1 2 3

Вопрос 10.

Сортировкой элементов массива будем называть такой способ обработки информации, который изменяет порядок элементов массива по следующему правилу: ai ≤ ai+1. Сортировка Шелла сравнивает элементы массива, отстоящие друг от друга на заданный интервал (шаг). В исходном массиве элементы располагались следующим образом:

2 5 7 8 1 9 2 4 6 1 5 4

Каким образом будут располагаться элементы массива после выполнения алгоритма с шагом 4?

1 1 2 4 2 5 5 4 6 9 7 8
1 1 2 2 4 4 5 5 6 7 8 9
1 2 1 4 2 5 4 5 6 9 7 8
1 5 7 8 2 9 2 4 6 1 5 4

Вопрос 11.

Сортировкой элементов массива будем называть такой способ обработки информации, который изменяет порядок элементов массива по следующему правилу: ai ≤ ai+1. Сортировка Хоара на каждой итерации алгоритма выбирает медианный элемент, а оставшиеся элементы массива переставляет следующим образом – все элементы, меньшие медианного записываются в левую часть массива, большие – в правую часть массива. В исходном массиве элементы располагались следующим образом:

2 5 7 8 1 9 2 4 6 1 5

Какое значение принимается в качестве медианного элемента, если реализована классическая версия алгоритма?

1
9
5
6

Вопрос 12.

В процессе обработки информации требуется построить двоичное представление целых чисел от 0 до 15, используя код Грея. Какой разряд инвертируется при получении 7-го набора? Примечание: нумерация разрядов начинается с 0. Первый набор 0001.

0
1
2
3

Вопрос 13

При обработке некоторой информации выполняется проверка включения множества A во множество B, алгоритм проверки использует принципы прямого слияния. Элементы множеств A и B перечислены ниже.

A = {0, 4, 7, 11, 18}
B = {0, 6, 7, 9, 10, 11, 17, 18}

Сколько сравнений выполнит алгоритм до окончательного решения?

1
2
3
4

Вопрос 14

Обрабатываемая информация представлена в виде множества X = {a,b,c,d,e}. Необходимо

Согласована на портале № 45603

сгенерировать все возможные сочетания из пяти имеющихся элементов множества по три элемента. Для решения этой задачи воспользовались алгоритмом генерации сочетаний в лексикографическом порядке. Какой из перечисленных наборов будет получен седьмым по счету?

{a, b, c}
{a, c, d}
{b, c, d}
{a, d, e}

Вопрос 15

Обрабатываемая информация представлена в виде множества X = {a,b,c,d,e}. Необходимо сгенерировать все возможные перестановки пяти имеющихся элементов. Для решения этой задачи воспользовались алгоритмом генерации перестановок в лексикографическом порядке. Каким по счету получен набор {a, c, d, b, e }?

4
6
7
9

Вопрос 16

Обрабатываемая информация представлена в виде множества X = {a,b,c,d,e}. Необходимо сгенерировать все возможные перестановки пяти имеющихся элементов. Для решения этой задачи воспользовались алгоритмом генерации перестановок транспозицией соседних элементов. Какой из перечисленных наборов будет получен пятым по счету?

{b, c, d, e, a}
{e, a, b, c, d}
{b, c, d, e, a}
{c, d, e, a, b}

Вопрос 17

Обрабатываемая информация представлена в виде множества X = {a,b,c,d,e}. Необходимо сгенерировать все возможные сочетания из пяти имеющихся элементов множества по три элемента. Для решения этой задачи воспользовались алгоритмом генерации сочетаний с помощью кодов Грея. Каким по счету получен набор {b, d, e}?

4
5
6
1

Вопрос 18

Что будет выведено на экран при выполнении следующего фрагмента программы при n =2500?

```
int func1(int p){
int s=0;
while(p>0){s+=p%10;
p/=10;
}
return s; }
int main(int argc, char *argv[])
{ int n;
scanf("%d",&n);
printf("%d\n",func1(n));
system("PAUSE");
return 0;}
```

1. На экран выведется сумма цифр числа.
2. На экран выведется максимальная цифра числа.
3. На экран выведется минимальная цифра числа.
4. На экран выведется количество цифр числа.

Вопрос 19

Согласована на портале № 45603

Чему будет равно значение переменной p при выполнении следующей программы, если элементы

массива y заданы следующим образом y = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}?

```
int Function1(int *x, int f, int l, int z) {
int S = 0,i;
for(i=f;i<l;i+=z) S+=x[i];
return S; }
int main(int argc, char *argv[])
{ int y[10],i,p;
for(i=0;i<10;i++)
y[i]=rand()%11-rand()%11;
p = Function1(y,0,10,1);
return 0; }
```

1. p - сумма всех элементов массива y.
2. p - сумма первых 5-ти элементов массива.
3. p - сумма элементов массива с четными индексами.
4. p - сумма элементов массива с нечетными индексами.

Вопрос 20

В процессе обработки информации о дорогах населенного пункта по карте был построен неориентированный граф: Для обработки этих данных на компьютере необходимо построить матрицу смежности этого графа. Какая последовательность нулей и единиц является первым столбцом этой матрицы?

1 0 0 0 1
1 1 0 0 0
0 1 0 0 1
1 0 1 0 0

14.1.2. Экзаменационные тесты

Приведены примеры типовых заданий из банка контрольных тестов, составленных по пройденным

разделам дисциплины.

Вопрос №1

Какую задачу решает представленный алгоритм?

```
алг Поиск нач
дано цел X[10]
цел n=10
цел a:=X[1]
для i от 2 до n нц
если X[i] < a то a:=X[i]
кц
рез a
кон
```

1. Поиск минимального элемента массива.
2. Поиск индекса первого минимального элемента массива.
3. Поиск индекса последнего минимального элемента массива.
4. Поиск максимального элемента массива.

Вопрос №2

Какую задачу решает представленный алгоритм?

```
алг Поиск нач
дано цел X[10]
цел n=10
цел b:=1
для i от 2 до n нц
если X[i]<X[b] то b:=I
кц
```

Согласована на портале № 45603

рез b
кон
1. Поиск минимального элемента массива.
2. Поиск индекса первого минимального элемента массива.
3. Поиск индекса последнего минимального элемента массива.
4. Поиск максимального элемента массива.
Вопрос №3
Какую задачу решает представленный алгоритм?
алг Поиск нач
дано цел X[10]
цел n=10
цел b:=1
для i от 2 до n нц
если X[i] ≤ X[b] то b:=I
кц
рез b
кон
1. Поиск минимального элемента массива.
2. Поиск индекса первого минимального элемента массива.
3. Поиск индекса последнего минимального элемента массива.
4. Поиск максимального элемента массива.
Вопрос №4
Какую задачу решает представленный алгоритм?
алг Вопрос4 нач
дано цел X[10]
цел n=10
цел a:=0
для i от 1 до n нц
если X[i] > 0 то a:=a+X[i]
кц
рез a
кон
1. Поиск суммы положительных элементов.
2. Поиск суммы индексов положительных элементов.
3. Поиск количества положительных элементов.
4. Поиск суммы отрицательных элементов.
Вопрос №5
Какую задачу решает представленный алгоритм?
алг Вопрос5 нач
дано цел X[10]
цел n=10
цел a:=0
для i от 1 до n нц
если X[i] > 0 то a:=a+i
кц
рез a
кон
1. Поиск суммы положительных элементов.
2. Поиск суммы индексов положительных элементов.
3. Поиск количества положительных элементов.
4. Поиск суммы отрицательных элементов.
Вопрос № 6
Программисту требовалось написать алгоритм поиска индекса первого элемента массива, имеющего положительное значение. Программист записал алгоритм следующим образом.

Согласована на портале № 45603

1. цел таб A[n]
2. для i от 1 до n нц
3. ввод A[i]
4. кц
5. для i от 1 до n нц
6. если A[i]>0 то p:=I
7. i:=n
8. кц
9. рез p

1. Алгоритм написан верно.
2. В строке 5.
3. В строке 6.
4. В строке 7.

Вопрос № 7

Программисту требовалось написать алгоритм поиска индекса первого элемента массива, имеющего положительное значение. Программист записал алгоритм следующим образом.

1. цел таб A[n]
2. для i от 1 до n нц
3. ввод A[i]
4. кц
5. для i от 1 до n нц
6. если A[i]>0 то p:=I
7. i:=i+1
8. кц
9. рез p

1. Алгоритм написан верно.
2. В строке 5.
3. В строке 6.
4. В строке 7..

Вопрос №8

Чему будет равно значение переменной p после выполнения алгоритма, если элементы массива

X ={1, 2, 3, 4, 5, 6, 7, 8, 9, 10} ?

n=10
цел таб X[n]
для i от 1 до n нц
ввод X[i]
кц
p:=1
для i от 2 до n нц
если X[p] > X[i] то p :=i
кц
рез p

1. Алгоритм находит индекс первого минимального элемента.
2. Алгоритм находит индекс последнего минимального элемента.
3. Алгоритм находит индекс первого максимального элемента.
4. Алгоритм находит индекс последнего максимального элемента.

Вопрос №9

Чему будет равно значение переменной p после выполнения алгоритма, если элементы массива

X ={-1, -2, -3, -4, -5, -6, -7, -8, -9, -10} ?

n=10
цел таб X[n]
для i от 1 до n нц

Согласована на портале № 45603

ввод X[i]
кц
p:=1
для i от 2 до n нц
если X[p] >= X[i] то p :=i
кц
рез p

1. Алгоритм находит индекс первого минимального элемента.
2. Алгоритм находит индекс последнего минимального элемента.
3. Алгоритм находит индекс первого максимального элемента.
4. Алгоритм находит индекс последнего максимального элемента.

Вопрос № 10.

Укажите описание соответствующее фрагменту программы:

```
int X[10];
int i;
for (i=0;i<10;i++)
X[i]=i+2;
```

1. Описание и инициализация статического целочисленного массива из 10-ти элементов.
2. Описание и инициализация динамического вещественного массива из 10 –ти элементов.
3. Описание и инициализация статического вещественного массива из 10-ти элементов.
4. Описание и инициализация динамического целочисленного массива из 10 –ти элементов.

Вопрос № 11.

Укажите описание соответствующее фрагменту программы:

```
int *X = (int*)malloc(sizeof(int)*10);
int I;
for (i=0;i<10;i++)
X[i]=i+2;
```

1. Описание и инициализация статического целочисленного массива из 10-ти элементов.
2. Описание и инициализация динамического вещественного массива из 10 –ти элементов.
3. Описание и инициализация статического вещественного массива из 10-ти элементов.
4. Описание и инициализация динамического целочисленного массива из 10 –ти элементов.

Вопрос № 12.

Выберите верные утверждения:

1. Оператор «->» - непрямой селектор компоненты.
2. Оператор «++» - бинарный оператор.
3. Оператор «%» - оператор целочисленного деления.
4. Оператор «=» - оператор «равно».

Вопрос № 13.

Выберите верные утверждения:

1. Оператор «++» - оператор префиксного (или постфиксного увеличения).
2. Оператор «^» - вычисление степени.
3. Оператор «%» - оператор целочисленного деления.
4. Оператор «*» - оператор прямой адресации.

Вопрос №14.

Выберите верные утверждения:

1. В последовательности int m; float b[12]; char* x; 13 лексем.
2. В последовательности int m; float b[12]; char* x; 11 лексем.
3. Последовательности float k,i; char z[12]; и float k, I; char z[12]; лексически не эквивалентны.
4. В последовательности int a,b; float * x; 7 лексем.

Вопрос №15

Чему будет равно значение переменной p после выполнения алгоритма, если элементы массива

X ={10, 9, 8, 7, 6, 5, 6, 7, 8, 2, 1} ?

Согласована на портале № 45603

n=10
цел таб X[n]
для i от 1 до n нц
ввод X[i]
кц
p:=1
для i от 2 до n нц
если X[p] < X[i] то p :=i
кц
рез p

1. Алгоритм находит индекс первого минимального элемента.
2. Алгоритм находит индекс последнего минимального элемента.
3. Алгоритм находит индекс первого максимального элемента.
4. Алгоритм находит индекс последнего максимального элемента.

Вопрос № 16

Выберите верные утверждения:

1. Алгоритм должен удовлетворять требованию конечности записи.
2. Алгоритм должен удовлетворять требованию последовательности записи.
3. Алгоритм должен удовлетворять требованию дискретности действий.
4. Алгоритм должен удовлетворять требованию структурности.

Вопрос № 17

Выберите верные утверждения:

1. Алгоритм должен содержать конечное количество шагов.
2. Алгоритм должен содержать заданное количество шагов.
3. Алгоритм должен выполнять заданное количество шагов при решении задачи.
4. Алгоритм должен быть дискретным для всех допустимых исходных данных.

Вопрос №18

Пяти программистам дали задания написать алгоритм суммирования целых чисел A1, A2, …, An. Некоторые программисты допустили ошибки в алгоритме. Какой фрагмент записан верно?

1.
цел S:=0
для I от A1 до An нц
S:=S+i
кц
рез S

2.
цел S:=A1
для I от A1 до An нц
S:=S+i
кц
рез S

3.
цел S:=0
i:=A1
пока I < An нц
S:=S+i
i:=i+1;
кц
рез S

4.
цел S:=0
i:=An
пока I ≥ A1 нц

Согласована на портале № 45603

S:=S+i
i:=i+1;
кц
рез S
Вопрос 19
Пяти программистам дали задания написать алгоритм произведения целых чисел A1, A2, …, An. Некоторые программисты допустили ошибки в алгоритме. Какой фрагмент записан верно?
1.
цел S:=1
для I от A1 до An нц
S:=S*i
кц
рез S
2.
цел S:=0
для I от A1 до An нц
S:=S*i
кц
рез S
3.
цел S:=1
i:=A1
пока I < An нц
S:=S*i
i:=i+1;
кц
рез S
4.
цел S:=0
i:=An
пока I ≥ A1 нц
S:=S*i
i:=i+1;
кц
рез S
Вопрос № 20
Выберите верно записанные идентификаторы:
1. sinus
2. 2Summa
3. Sinu*s
4. double

14.1.3. Темы контрольных работ

Программирование
Вопрос № 1
Что будет выведено на экран при выполнении следующего фрагмента программы
int i;
for(i=1;i<10;i+=2) printf("%d ",i);
1. Вывод на экран чисел 1, 3, 5, 7, 9
2. Вывод на экран чисел 1, 4, 9, 16, 25
3. Вывести на экран числа 3,5,7,9,11
4. Вывод на экран чисел 3,5,7,9,11
Вопрос № 2
Что будет выведено на экран при выполнении следующего фрагмента программы

Согласована на портале № 45603

```
int i;
for(i=0;i<5;i++) printf("%d ",(i+1)*(i+1));
```
1. Вывод на экран чисел 1, 3, 5, 7, 9
2. Вывод на экран чисел 1, 4, 9, 16, 25
3. Вывести на экран числа 3,5,7,9,11
4. Вывод на экран чисел 3,5,7,9,11

Вопрос № 3

Выберите фрагменты программ, в которых цикл выполняется 7 раз.

1. `int i=1; while (i<20) i+=3;`
2. `int i=0; while (i<15)i+=3;`
3. `int i=0; while (i<12)i+=2;`
4. `int i=4; while (i<40) i+=i;`

Вопрос № 4

Что будет выведено на экран при выполнении следующего фрагмента программы при n = 2500?

```
int func1(int p){
int s=0;
while(p>0){s+=p%10;
p/=10;
}
return s; }
int main(int argc, char *argv[])
{ int n;
scanf("%d",&n);
printf("%d\n",func1(n));
system("PAUSE");
return 0;}
```

1. На экран выведется сумма цифр числа.
2. На экран выведется максимальная цифра числа.
3. На экран выведется минимальная цифра числа.
4. На экран выведется количество цифр числа.

Вопрос №5

Что будет выведено на экран при выполнении следующего фрагмента программы при n = 1000 , m = 5?

```
int func13(int p, int v){
int s=0,t;
while(p>0){t=p%10;
if(t==v)s++;
p/=10;
}
return s; }
int main(int argc, char *argv[])
{ int n,m;
scanf("%d%d",&n,&m);
printf("%d\n",func13(n,m));
system("PAUSE");
return 0;}
```

1. На экран выведется количество цифр числа n равных m.
2. На экран выведется количество цифр числа n не равных m.
3. На экран выведется количество цифр числа n больших m.
4. На экран выведется количество цифр числа n меньших m.

Вопрос №6

Что будет выведено на экран при выполнении следующего фрагмента программы при n = 2020 , m =7?

Согласована на портале № 45603

```
int func16(int p, int v){
int s=0,t;
while(p>0){t=p%10;
if(t<v)s++;
p/=10;
}
return s; }
int main(int argc, char *argv[])
{ int n,m;
scanf("%d%d",&n,&m);
printf("%d\n",func16(n,m));
system("PAUSE");
return 0;}
```

1. На экран выведется количество цифр числа n равных m.
2. На экран выведется количество цифр числа n не равных m.
3. На экран выведется количество цифр числа n больших m.
4. На экран выведется количество цифр числа n меньших m.

Вопрос №7

Чему будет равно значение переменной p при выполнении следующей программы, если элементы массива y заданы следующим образом y = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}?

```
int Function1(int *x, int f, int l, int z) {
int S = 0,i;
for(i=f;i<l;i+=z) S+=x[i];
return S; }
int main(int argc, char *argv[])
{ int y[10],i,p;
for(i=0;i<10;i++)
y[i]=rand()%11-rand()%11;
p = Function1(y,0,10,1);
return 0; }
```

1. p - сумма всех элементов массива y.
2. p - сумма первых 5-ти элементов массива.
3. p - сумма элементов массива с четными индексами.
4. p - сумма элементов массива с нечетными индексами.

Вопрос №8

Чему будет равно значение переменной p при выполнении следующей программы, если элементы массива y заданы следующим образом y = {-10, 2, 3, 4, 5, 6, 7, 8, 9, 1}?

```
p = Function1(y,1,20,2);
int Function1(int *x, int f, int l, int z) {
int S = 0,i;
for(i=f;i<l;i+=z) S+=x[i];
return S; }
int main(int argc, char *argv[])
{ int y[10],i,p;
for(i=0;i<10;i++)
y[i]=rand()%11-rand()%11;
p = Function1(y,1,10,2);
return 0;}
```

1. p - сумма всех элементов массива y.
2. p - сумма первых 5-ти элементов массива.
3. p - сумма элементов массива с четными индексами.
4. p - сумма элементов массива с нечетными индексами.

Согласована на портале № 45603

Вопрос №9

Чему будет равно значение переменной p при выполнении следующей программы, если элементы

массива y заданы следующим образом y = {-10, 2, 3, 4, 5, 6, 7, 8, 9, 1}?

```
p = Function1(y,1,20,2);
int Function1(int *x, int f, int l, int z) {
int S = 0,i;
for(i=f;i<l;i+=z) S+=x[i];
return S; }
int main(int argc, char *argv[])
{ int y[10],i,p;
for(i=0;i<10;i++)
y[i]=rand()%11-rand()%11;
p = Function1(y,1,10,2);
return 0;}
```

1. p - сумма всех элементов массива y.
2. p - сумма первых 5-ти элементов массива.
3. p - сумма элементов массива с четными индексами.
4. p - сумма элементов массива с нечетными индексами.

Вопрос №10

Сколько шагов выполнит внешний шаг сортировки обменом при сортировке элементов массива

X = {-5, 3, 0, -7, 10, -3, -5, 1, 2, 5}?

```
int main() {
int n = 10, X[n],i,C=0,p=0,v,j;
srand(time(NULL));
for(i=0;i<10;i++){
X[i]=rand()%11-rand()%11;
printf("%d ",X[i]);}
for(i=0;i<n-1;i++){
p=0;C++;
for(j=0;j<n-i-1;j++)
{ if(X[j]>X[j+1])
{ v=X[j];
X[j]=X[j+1];
X[j+1]=v;
p++;}}
if (p==0)break; }
printf("\n");
for(i=0;i<10;i++)
printf("%d ",X[i]);
printf("\n %d \n",C);
system("PAUSE");
return 0;
}
```

1. 10
2. 8
3. 5
4. 6

14.1.4. Темы лабораторных работ

Проверка условий

Обработка массивов

14.1.5. Методические рекомендации

Учебный материал излагается в форме, предполагающей самостоятельное мышление

Согласована на портале № 45603

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 необходимо осуществлять медленно, выделяя основные идеи; на основании изученного составить тезисы. Освоив материал, попытаться соотнести теорию с примерами из практики;
- если в тексте встречаются термины, следует выяснить их значение для понимания дальнейшего материала;
- необходимо осмысливать прочитанное и изученное, отвечать на предложенные вопросы.

Студенты могут получать индивидуальные консультации с использованием средств телекоммуникации.

По дисциплине могут проводиться дополнительные занятия в форме вебинаров. Расписание вебинаров публикуется в кабинете студента на сайте Университета. Запись вебинара публикуется в электронном курсе по дисциплине.

14.2. 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 14.

Таблица 14 – 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
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 исходя из состояния обучающегося на момент проверки

14.3. 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;

Согласована на портале № 45603

– 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

Согласована на портале № 45603