

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Алейник Станислав Николаевич  
Должность: Ректор  
Дата подписания: 08.04.2021 18:21:19  
Уникальный программный ключ:  
5258223550ea9fbeb23726a1609b644b33d8986ab6255891f288f913a1351fae

1

**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ**  
**ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ**  
**имени В.Я.ГОРИНА»**

**УТВЕРЖДАЮ**

Декан инженерного факультета  
профессор  С.В. Стребков



«07» 07 2020 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**РАЗРАБОТКА ПРОГРАММНЫХ ПРИЛОЖЕНИЙ**

Направление подготовки: 09.03.03 - Прикладная информатика

Направленность (профиль): Прикладная информатика в АПК

Квалификация: бакалавр

Год начала подготовки: 2020

Майский, 2020

Рабочая программа дисциплины (модуля) составлена с учетом требований:

- федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.03 – Прикладная информатика, утвержденного приказом Министерства образования и науки РФ от 19 сентября 2017 г. № 922;
- порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры, утвержденного приказом Министерства образования и науки РФ от 05.04.2017 г., № 301;
- профессионального стандарта «Специалист по информационным системам», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2014 г. № 896н
- профессионального стандарта «Системный аналитик», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 28 октября 2014 г. № 809н

**Составители:** к.т.н., доцент Игнатенко В.А.

**Рассмотрена** на заседании кафедры информатики и информационных технологий

« 18 » 06 2020 г., протокол № 73

И.о. зав. кафедрой  Е.В. Голованова

Руководитель основной профессиональной образовательной программы  В.А. Игнатенко

## I. ЦЕЛЬ И ЗАДАЧИ ДИСЦИПЛИНЫ

**1.1. Цель дисциплины** – обучить студентов созданию, отладке и тестированию программных приложений в интегрированной среде разработки.

### 1.2. Задачи:

- совершенствовать знания объектно-ориентированного и визуального программирования, алгоритмов компьютерной обработки структур данных, а также технологии программирования;
- развить профессиональные компетенции, включая технологию разработки программного обеспечения на языках высокого уровня.

## II. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОСНОВНОЙ ПРОФЕССИОНАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ (ОПОП)

### 2.1. Цикл (раздел) ОПОП, к которому относится дисциплина

Дисциплина «Разработка программных приложений» относится к дисциплинам вариативной части (Б1.В.02) основной профессиональной образовательной программы

### 2.2. Логическая взаимосвязь с другими частями ОПОП

<b>Наименование предшествующих дисциплин, практик, на которых базируется данная дисциплина (модуль)</b>	1. Математика
	2. Информатика и программирование
	3. Информационные системы и технологии
<b>Требования к предварительной подготовке обучающихся -</b>	<p><i>знать:</i> основы теории информации; основные программные конструкции; основы концепций функционального и объектно-ориентированного программирования.</p> <p><i>уметь:</i> реализовывать программные алгоритмы на языке PASCAL, Си</p> <p><i>владеть:</i> Интегрированными средами программирования и отладки</p>

Дисциплина является предшествующей для дисциплин: «Проектный практикум», «Программирование информационных систем».

### III. ОБРАЗОВАТЕЛЬНЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДИСЦИПЛИНЫ, СООТВЕТСТВУЮЩИЕ ФОРМИРУЕМЫМ КОМПЕТЕНЦИЯМ

Коды компетенций	Формулировка компетенции	Индикаторы Достижения компетенции	Планируемые результаты обучения по дисциплине
ПК-1	Способность проводить описание прикладных процессов и информационного обеспечения решения прикладных задач	<p>ПК-1.1 Демонстрирует и использует знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов</p> <p>ПК-1.3 Демонстрирует навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов</p>	<p><b>Знать:</b> основные методологии описания архитектуры вычислительной системы и программных алгоритмов</p> <p><b>Уметь:</b> настраивать программное обеспечение для работы в сети Интернет;</p> <p><b>Владеть:</b> основами методологий описания архитектуры вычислительной системы и программных алгоритмов</p> <p><b>Знать:</b> прикладные задачи с использованием методологий описания программных алгоритмов</p> <p><b>Уметь:</b> демонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов</p> <p><b>Владеть:</b> навыками демонстрации формализации прикладной задачи с исполь-</p>

ПК-3	Способен применять в профессиональной деятельности современные языки программирования и моделирования	<p>ПК-3.1 Использует объектно-ориентированную парадигму средств программирования и моделирования</p> <p>ПК-3.2 Демонстрирует навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p> <p>ПК-3.3 Демонстрирует навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки</p>	<p>зованием методологий описания программных алгоритмов</p> <p><b>Знать:</b> объектно-ориентированную парадигму средств программирования и моделирования <b>Уметь:</b> применять объектно-ориентированную парадигму средств программирования и моделирования <b>Владеть:</b> навыками использования объектно-ориентированной парадигмы средств программирования и моделирования</p> <p><b>Знать:</b> принципы построения, программирования и эксплуатации систем с использованием микропроцессорной техники; <b>Уметь:</b> применять принципы построения, программирования и эксплуатации систем с использованием микропроцессорной техники; <b>Владеть:</b> навыками построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p> <p><b>Знать:</b> принципы алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки <b>Уметь:</b> применять навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки <b>Владеть:</b> навыками алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки</p>
------	-------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

--	--	--	--

#### IV. ОБЪЕМ, СТРУКТУРА, СОДЕРЖАНИЕ ДИСЦИПЛИНЫ, ВИДЫ УЧЕБНОЙ РАБОТЫ И ФОРМЫ КОНТРОЛЯ ЗНАНИЙ

##### 4.1 Распределение объема учебной работы по формам обучения

Вид работы (в соответствии с учебным планом)	Объем учебной работы, час	
	Очная	Заочная
<b>Формы обучения</b> (вносятся данные по реализуемым)		
<b>Семестр изучения дисциплины</b>	<b>4</b>	<b>2</b>
Общая трудоемкость, всего, час	144	144
зачетные единицы	4	4
<b>1. Контактная работа</b>		
<b>1.1 Контактная аудиторная работа (всего)</b>	<b>40,25</b>	<b>16,45</b>
В том числе:		
Лекции ( <i>Лек</i> )	20	4
Лабораторные занятия ( <i>Лаб</i> )	20	4
Практические занятия ( <i>Пр</i> )		
Установочные занятия ( <i>УЗ</i> )		2
Предэкзаменационные консультации ( <i>Конс</i> )		
Текущие консультации ( <i>ТК</i> )		6
<b>1.2 Промежуточная аттестация</b>		
Зачет ( <i>КЗ</i> )	0,2	,2
Экзамен ( <i>КЭ</i> )		
Выполнение курсовой работы (проекта) ( <i>КНKP</i> )		0,2
<b>1.3 Контактная внеаудиторная работа (контроль)</b>	<b>20</b>	<b>4</b>
<b>в том числе по семестрам</b>	18	4
<b>2. Самостоятельная работа обучающихся (всего)</b>	<b>83,75</b>	<b>123,55</b>
в том числе:		
Самостоятельная работа по проработке лекционного материала	12	2
Самостоятельная работа по подготовке к лабораторно-практическим занятиям	12	2
Работа над темами (вопросами), вынесенными на самостоятельное изучение	41,75	121,55
Самостоятельная работа по видам индивидуальных заданий: подготовка реферата (контрольной работы)	10	10
Подготовка к зачету	8	8

## 4.2 Общая структура дисциплины и виды учебной работы

Наименование модулей и разделов дисциплины	Объемы видов учебной работы по формам обучения, час							
	Очная форма обучения				Заочная форма обучения			
	Всего	Лекции	Лабораторно-практич. занятия	Самостоятельная работа	Всего	Лекции	Лабораторно-практич. занятия	Самостоятельная работа
1	2	3	4	6	7	8	9	11
<b>7 семестр</b>				<b>4 курс , летняя сессия</b>				
<b>Модуль 1</b>	<b>60</b>	<b>10</b>	<b>10</b>	<b>40</b>	<b>64</b>	<b>2</b>	<b>2</b>	<b>60</b>
1. Введение в язык СИ. Структура программы.	9	2	1	6	12,8	0,4	0,4	12
2. Объекты языка Си и их типы. Простые объекты	12	2	2	8	12,8	0,4	0,4	12
3. Операции. Ввод и вывод информации. Операторы.	12	2	2	8	12,8	0,4	0,4	12
4. Функции. Прототипы функций.	12	2	2	8	12,8	0,4	0,4	12
5. Препроцессор. Глобальные и локальные объекты. Математические функции.	9	2	1	8	12,8	0,4	0,4	12
<i>Итоговое занятие по модулю 1</i>	4	-	2	2	0	-	-	-
<b>Модуль 2</b>	<b>63,75</b>	<b>10</b>	<b>10</b>	<b>43,75</b>	<b>67,55</b>	<b>2</b>	<b>2</b>	<b>63,55</b>
1. Указатели. Модели памяти. Динамическое распределение памяти	11	2	1	8	12,8	0,4	0,4	12
2. Массивы. Передача массива	12	2	2	8		0,4	0,4	12
3. Динамическое распределение памяти под массивы	12	2	2	8	12,8	0,4	0,4	12
4. Структуры. Объединения.	12	2	2	8	12,8	0,4	0,4	12
5. Функции ввода-вывода. Работа с файлами данных. Работа со строками	12,75	2	1	9,75	16,35	0,4	0,4	15,55
<i>Итоговое занятие по модулю 2</i>	4	-	2	2	0	-	-	-
<i>Предэкзаменационные консультации</i>								
<i>Текущие консультации</i>					6			
<i>Зачет</i>	0,25				0,25			
<b>ИТОГО:</b>								
<i>Контактная аудиторная работа</i>	40,25				16,45			
<i>Контактная внеаудиторная работа</i>	20				4			
<i>Самостоятельная работа</i>	83,75				123,55			

<b>Наименование и содержание модулей и разделов дисциплины</b>
<b>Модуль 1</b>
1. Введение в язык СИ. Структура программы.
2. Объекты языка Си и их типы. Простые объекты
3. Операции. Ввод и вывод информации. Операторы.
4. Функции. Прототипы функций.
5. Препроцессор. Глобальные и локальные объекты. Математические функции.
<b>Модуль 2</b>
1. Указатели. Модели памяти. Динамическое распределение памяти
2. Массивы. Передача массива
3. Динамическое распределение памяти под массивы
4. Структуры. Объединения.
5. Функции ввода-вывода. Работа с файлами данных. Работа со строками

## V. ОЦЕНКА ЗНАНИЙ И ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ ЗНАНИЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

### 5.1 Формы контроля знаний, рейтинговая оценка и формируемые компетенции (дневная форма обучения)

Наименование рейтингов, модулей и блоков	Формируемые компетенции	Объем учебной работы				Форма контроля знаний	Количество баллов (min)	Количество баллов (max)
		Общая трудоёмкость	Лекции	Лаборат.-практические занятия	Самостоятельная работа			
<b>Всего по дисциплине</b>	ПК-1 ПК-3	144	42	56	170, 6	зачет	51	100
<i>I Входной рейтинг</i>						Тестирова-	5	5
<i>II Рубежный рейтинг</i>						Сумма бал- лов за мо- дули	51	60
<b>Модуль 1</b>	ПК-1 ПК-3	<b>60</b>	<b>10</b>	<b>10</b>	<b>40</b>		10	30
1. Введение в язык СИ. Структура программы.		9	2	1	6	Устный опрос, решение задач		
2. Объекты языка Си и их типы. Простые объекты		12	2	2	8	Устный опрос, решение задач		
3. Операции. Ввод и вывод информации. Операторы.		12	2	2	8	Устный опрос, решение задач		
4. Функции. Прототипы функций.		12	2	2	8			
5. Препроцессор. Глобальные и локальные объекты. Математические функции.		9	2	1	8			
Итоговый контроль знаний по темам модуля 1		4	-	2	2	Тестирова- ние, си- туацион- ные зада- чи		
<b>Модуль 2</b>	ПК-1 ПК-3	<b>63,7 5</b>	<b>10</b>	<b>10</b>	<b>43,7 5</b>		21	30
1. Указатели. Модели памяти. Динамическое распределение памяти		11	2	1	8	Устный опрос, ре-		

						шение за- дач		
2. Массивы. Передача массива		12	2	2	8	Устный опрос, ре- шение за- дач		
3. Динамическое распределение памяти под массивы		12	2	2	8	Устный опрос, ре- шение за- дач		
4. Структуры. Объединения.		12	2	2	8			
5. Функции ввода-вывода. Работа с файлами данных. Работа со строками		12,7 5	2	1	9,75			
Итоговый контроль знаний по темам модуля 2		4	-	2	2	Тестиро- вание, си- туацион- ные зада- чи		
Контрольная работа							2	2
Творческий рейтинг						реферат	3	3
Выходной рейтинг						зачет	15	25

## 5.2 Оценка знаний студента

### 5.2.1. Основные принципы рейтинговой оценки знаний

Оценка знаний по дисциплине осуществляется согласно Положению о балльно-рейтинговой системе оценки обучения в ФГБОУ Белгородского ГАУ.

Уровень развития компетенций оценивается с помощью рейтинговых баллов.

Рейтинги	Характеристика рейтингов	Максимум баллов
Рубежный	Отражает работу студента на протяжении всего периода изучения дисциплины. Определяется суммой баллов, которые студент получит по результатам изучения каждого модуля.	60
Творческий	Результат выполнения студентом индивидуального творческого задания различных уровней сложности, в том числе, участие в различных конференциях и конкурсах на протяжении всего курса изучения дисциплины.	5
Рейтинг личностных качеств	Оценка личностных качеств обучающихся, проявленных ими в процессе реализации дисциплины (модуля) (дисциплинированность, посещаемость учебных занятий, сдача вовремя контрольных мероприятий, ответствен-	10

	ность, инициатива и др.)	
Рейтинг сформированности прикладных практических требований	Оценка результата сформированности практических навыков по дисциплине (модулю), определяемый преподавателем перед началом проведения промежуточной аттестации и оценивается как «зачтено» или «не зачтено».	+
Промежуточная аттестация	Является результатом аттестации на окончательном этапе изучения дисциплины по итогам сдачи зачета или экзамена. Отражает уровень освоения информационно-теоретического компонента в целом и основ практической деятельности в частности.	25
Итоговый рейтинг	Определяется путём суммирования всех рейтингов	100

Итоговая оценка компетенций студента осуществляется путём автоматического перевода баллов общего рейтинга в стандартные оценки:

Не зачтено	Зачтено	Зачтено	Зачтено
менее 51 балла	51-67 баллов	67,1-85 баллов	85,1-100 баллов

*Если форма контроля «экзамен»:*

### **5.2.2. Критерии оценки знаний студента на экзамене**

На экзамене студент отвечает в письменно-устной форме на вопросы экзаменационного билета (2 вопроса и задача).

Количественная оценка на экзамене определяется на основании следующих критериев:

- оценку «отлично» заслуживает студент, показавший всестороннее систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой; как правило, оценка «отлично» выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины и их значение для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;
- оценку «хорошо» заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе; как правило, оценка «хорошо» выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности;
- оценку «удовлетворительно» заслуживает студент, обнаруживший знания основного учебно-программного материала

в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой; как правило, оценка «удовлетворительно» выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя;

- оценка «неудовлетворительно» выставляется студенту, обнаружившему проблемы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий; как правило, оценка «неудовлетворительно» ставится студентам, которые не могут продолжать обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

*Если форма контроля «зачет»:*

### **5.2.2. Критерии оценки знаний студента на зачете**

Оценка «зачтено» на зачете определяется на основании следующих критериев:

- студент усвоил взаимосвязь основных понятий дисциплины и их значение для приобретаемой профессии, при этом проявил творческие способности в понимании, изложении и использовании учебно-программного материала;
- студент демонстрирует полное знание учебно-программного материала, успешно выполнил предусмотренные в программе задания, усвоил основную литературу, рекомендованную в программе;
- студент показал систематический характер знаний по дисциплине и способность к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.

Оценка «не зачтено» на зачете определяется на основании следующих критериев:

- студент допускает грубые ошибки в ответе на зачете и при выполнении заданий, при этом не обладает необходимыми знаниями для их устранения под руководством преподавателя;
- студент демонстрирует проблемы в знаниях основного учебно-программного материала, допускает принципиальные ошибки в выполнении предусмотренных программой заданий;
- студент не может продолжать обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

**5.3. Фонд оценочных средств. Типовые контрольные задания или иные материалы, необходимые для оценки формируемых компетенций по дисциплине (приложение 1)**

## **VI. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **6.1. Основная учебная литература**

1. Васюткина, И.А. Разработка приложений на С# с использованием СУБД PostgreSQL [Электронный ресурс] / Васюткина И.А., Трошина Г.В., Бычков М.И. - Новосиб.:НГТУ, 2015. - 143 с. Режим доступа: <http://znanium.com/bookread2.php?book=556925>

### **6.2. Дополнительная литература**

1. Технология разработки программного обеспечения: Учеб. пос. [Электронный ресурс] / Л.Г.Гагарина, Е.В.Кокорева, Б.Д.Виснадул; Под ред. проф. Л.Г.Гагариной - М.: ИД ФОРУМ: НИЦ Инфра-М, 2013. - 400 с. Режим доступа: <http://znanium.com/bookread2.php?book=389963>

2. Игнатенко, В.А. Методические указания и задания к выполнению лабораторно – практических работ студентов по дисциплине «Разработка про-граммных приложений»/ сост. В.А. Игнатенко - Белгород: Белгородский ГАУ, 2014. – 48с.

Режим доступа: <https://clck.ru/FDrZr>

### **6.3 Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине**

Самостоятельная работа студентов заключается в инициативном поиске информации о наиболее актуальных проблемах, которые имеют большое практическое значение и являются предметом научных дискуссий в рамках изучаемой дисциплины.

Самостоятельная работа планируется в соответствии с календарными планами рабочей программы по дисциплине и в методическом единстве с те-матикой учебных аудиторных занятий.

Самостоятельную работу студента поддерживает электронная информационная среда ВУЗа, доступ к которой <http://do.belgau.edu.ru> (логин, пароль студента)

### **6.3.1 Методические указания по освоению дисциплины**

Вид учебных занятий	Организация деятельности студента
1	2
Лекции	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные определения, теоремы, основные задачи, методы решений задач, выводы, обобщения; пометать важные мысли, выделять ключевые слова, термины. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации или практическом занятии.
Лабораторно-практические занятия	Проработка рабочей программы, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины. Работа с конспектом лекций, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы (см. п.6.1 и 6.2). Решение задач по теме занятия, выполнение расчетно-графических заданий.
Самостоятельная работа	Изучение теоретического материала по конспекту лекций, знакомство с основной и дополнительной литературой, включая справочные издания, Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Решение задач по темам практических занятий, выполнение расчетно-графических заданий.
Подготовка к экзамену	При подготовке к зачету и экзамену необходимо руководствоваться конспектом лекций, материалами практических занятий, рекомендуемой литературой, а также перечнем экзаменационных вопросов и типовыми контрольными тестами (см. приложение).

### **6.3.1. Методические указания по освоению дисциплины**

1. Игнатенко, В.А. Методические указания по самостоятельной работе студентов [Электронный ресурс]/ В.А. Игнатенко, В.Л. Михайлова// Изд. Белгородский ГАУ. 2015. - 42 с.

### **Печатные периодические издания**

1. Журнал «[Информационные технологии](#)»
2. Журнал «Вестник российской сельскохозяйственной науки»
3. Журнал «Достижения науки и техники АПК»
4. Журнал «Экономика, статистика и информатика»

#### **6.4. Ресурсы информационно-телекоммуникационной сети «Интернет», современные профессиональные базы данных, информационные справочные системы.**

1. Электронные ресурсы по математике <http://lbz.ru/metodist/iumk/mathematics/er.php>
2. Общероссийский математический портал (информационная система) - <http://www.mathnet.ru/>
3. Профессиональная база данных и информационно справочная система по официальной технической документации для разработчиков под ОС [Microsoft Windows](https://msdn.microsoft.com/ru-ru) <https://msdn.microsoft.com/ru-ru>
4. Профессиональная база данных и информационно справочная система по официальной технической документации для разработчиков под ОС [Microsoft Windows](https://technet.microsoft.com/ru-ru) <https://technet.microsoft.com/ru-ru>

#### **6.5. Перечень программного обеспечения, информационных технологий.**

1. Операционная система Windows;
2. Пакет программ Microsoft Office;
3. SunRav – программа для тестирования;
4. PIPE 3.2 (среда моделирования сетей Петри, распространяется бесплатно)

## **VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **7.1. Помещения, укомплектованные специализированной мебелью, оснащенные оборудованием и техническими средствами обучения, служащими для представления учебной информации большой аудитории**

<b>Виды помещений</b>	<b>Оборудование и технические средства обучения</b>
Учебная аудитория для проведения занятий лекционного типа № .	Специализированная мебель для обучающихся на посадочных мест. Рабочее место преподавателя: стол, стул, кафедра-трибуна напольная, доска меловая настенная. Набор демонстрационного оборудования: Ноутбук , проектор , экран для демон-

	страции, 2 акустические колонки. Информационные стенды (планшеты настенные):
Учебная аудитория для проведения занятий лекционного типа, семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации №.	Специализированная мебель для обучающихся на посадочных мест. Рабочее место преподавателя: стол, стул, кафедра-трибуна напольная, доска меловая настенная. Набор демонстрационного оборудования: - проектор _____ ; - экран для проектора; - 2 акустические колонки - ноутбук _____. Информационные стенды (планшеты настенные): - - - -
Помещения для самостоятельной работы обучающихся с возможностью подключения к Интернету и обеспечением доступа в электронную информационно-образовательную среду Белгородского ГАУ (читальные залы библиотеки)	Специализированная мебель; комплект компьютерной техники в сборе (системный блок: Asus P4BGL-MX\Intel Celeron, 1715 MHz\256 Мб PC2700 DDR SDRAM\ST320014A (20 Гб, 5400 RPM, Ultra-ATA/100)\ NEC CD-ROM CD-3002A\Intel(R) 82845G/GL/GE/PE/GV Graphics Controller, монитор: Proview 777(N) / 786(N) [17" CRT], клавиатура, мышь.) в количестве 10 единиц с возможностью подключения к сети Интернет и обеспечения доступа в электронную информационнообразовательную среду Белгородского ГАУ; настенный плазменный телевизор SAMSUNG PS50C450B1 Black HD (диагональ 127 см); аудиовидео кабель HDMI
Помещение для хранения и профилактического обслуживания учебного оборудования №	Специализированная мебель: Рабочее место лаборанта:

## 7.2. Комплект лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства

Виды помещений	Оборудование
Учебная аудитория для проведения занятий лекционного типа № .	MS Windows WinStrtr 7 Acdmc Legalization RUS OPL NL. Договор №180 от 12.02.2011.

	Срок действия лицензии – бессрочно; MS Office Std 2010 RUS OPL NL Acdmc. Договор №180 от 12.02.2011. Срок действия лицензии – бессрочно; Anti-virus Kaspersky Endpoint Security для бизнеса (Сублицензионный договор №28 от 08.11.2018) - 522 лицензия. Срок действия лицензии с 08.11.2018 по 08.11.2019
Учебная аудитория для проведения занятий лекционного типа, семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации №	MS Windows WinStrtr 7 Acdmc Legalization RUS OPL NL. Договор №180 от 12.02.2011. Срок действия лицензии – бессрочно; MS Office Std 2010 RUS OPL NL Acdmc. Договор №180 от 12.02.2011. Срок действия лицензии – бессрочно; Anti-virus Kaspersky Endpoint Security для бизнеса (Сублицензионный договор №28 от 08.11.2018) - 522 лицензия. Срок действия лицензии с 08.11.2018 по 08.11.2019
Помещения для самостоятельной работы обучающихся с возможностью подключения к Интернету и обеспечением доступа в электронную информационно-образовательную среду Белгородского ГАУ (читальные залы библиотеки)	Microsoft Imagine Premium Electronic Software Delivery. Сублицензионный договор №937/18 на передачу неисключительных прав от 16.11.2018. Срок действия лицензии- бессрочно. MS Office Std 2010 RUSOPLNL Acdmc. Договор №180 от 12.02.2011. Срок действия лицензии – бессрочно. Anti-virus Kaspersky Endpoint Security для бизнеса (Сублицензионный договор №28 от 08.11.2018).Срок действия лицензии с 08.11.2018 по 08.11.2019 Информационно правовое обеспечение "Гарант" (для учебного процесса). Договор №ЭПС-12-119 от 01.09.2012. Срок действия - бессрочно. СПС КонсультантПлюс: Версия Проф. Консультант Финансист. КонсультантПлюс: Консультации для бюджетных организаций. Договор от 01.01.2017. Срок действия - бессрочно. RHVoice-v0.4-a2 синтезатор речи Программа Balabolka (portable) для чтения вслух текстовых файлов. Программа экранного доступа NDVA
Помещение для хранения и профилактического обслуживания учебного оборудования №	MS Windows WinStrtr 7 Acdmc Legalization RUS OPL NL. Договор №180 от 12.02.2011. Срок действия лицензии – бессрочно; MS Office Std 2010 RUS OPL NL Acdmc. Договор №180 от 12.02.2011. Срок действия лицензии – бессрочно; Anti-virus Kaspersky Endpoint Security для бизнеса (Сублицензионный договор №28 от 08.11.2018) - 522 лицензия. Срок действия лицензии с 08.11.2018 по 08.11.2019

## **VIII. ОСОБЕННОСТИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ) ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ**

В случае обучения в университете инвалидов и лиц с ограниченными возможностями здоровья учитываются особенности психофизического развития, индивидуальные возможности и состояние здоровья таких обучающихся.

Образование обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах. Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечены печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья. Обучение инвалидов осуществляется также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).

Для лиц с ограниченными возможностями здоровья по слуху возможно предоставление учебной информации в визуальной форме (краткий конспект лекций; тексты заданий). На аудиторных занятиях допускается присутствие ассистента, а также сурдопереводчиков и (или) тифлосурдопереводчиков. Текущий контроль успеваемости осуществляется в письменной форме: обучающийся письменно отвечает на вопросы, письменно выполняет практические задания. Доклад (реферат) также может быть представлен в письменной форме, при этом требования к содержанию остаются теми же, а требования к качеству изложения материала (понятность, качество речи, взаимодействие с аудиторией и т. д.) заменяются на соответствующие требования, предъявляемые к письменным работам (качество оформления текста и списка литературы, грамотность, наличие иллюстрационных материалов и т.д.). Промежуточная аттестация для лиц с нарушениями слуха проводится в письменной форме, при этом используются общие критерии оценивания. При необходимости время подготовки к ответу может быть увеличено.

Для лиц с ограниченными возможностями здоровья по зрению университетом обеспечивается выпуск и использование на учебных занятиях альтернативных форматов печатных материалов (крупный шрифт или аудиофайлы) а также обеспечивает обучающихся надлежащими звуковыми средствами воспроизведения информации (диктофонов и т.д.). Допускается присутствие ассистента, оказывающего обучающемуся необходимую техническую помощь. Текущий контроль успеваемости осуществляется в устной форме. При проведении промежуточной аттестации для лиц с нарушением зрения тестирование может быть заменено на устное собеседование по вопросам.

Для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно- двигательного аппарата материально-технические условия уни-

верситета обеспечивают возможность беспрепятственного доступа обучающихся в учебные помещения, а также пребывания в них (наличие пандусов, поручней, расширенных дверных проемов, лифтов; наличие специальных кресел и других приспособлений). На аудиторных занятиях, а также при проведении процедур текущего контроля успеваемости и промежуточной аттестации лицам с ограниченными возможностями здоровья, имеющим нарушения опорно-двигательного аппарата могут быть предоставлены необходимые технические средства (персональный компьютер, ноутбук или другой гаджет); допускается присутствие ассистента (ассистентов), оказывающего обучающимся необходимую техническую помощь (занять рабочее место, передвигаться по аудитории, прочитывать задание, оформить ответ, общаться с преподавателем).



МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕ-  
РАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬ-  
НОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «БЕЛГОРОДСКИЙ ГОСУ-  
ДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ  
имени В.Я.ГОРИНА»

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
**для проведения промежуточной аттестации обучающихся**  
по дисциплине **«Разработка программных приложений»**

Направление подготовки/специальность: 09.03.03 – Прикладная информатика

Направленность (профиль): Прикладная информатика в АПК

Квалификация: бакалавр

Год начала подготовки: 2020

Майский, 2020

## 1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Код контролируемой компетенции	Формулировка контролируемой компетенции	Индикаторы достижения компетенции	Этап (уровень) освоения компетенции	Планируемые результаты обучения	Наименование модулей и (или) разделов дисциплины	Наименование оценочного средства	
						Текущий контроль	Промежуточная аттестация
ПК-1	Способность проводить описание прикладных процессов и информационного обеспечения решения прикладных задач	<p>ПК-1.1 Демонстрирует и использует знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов</p> <p>ПК-1.3 Демонстрирует навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов</p>	Первый этап (пороговой уровень)	<p><b>Знать:</b> основные методологии описания архитектуры вычислительной системы и программных алгоритмов</p> <p><b>Знать:</b> прикладные задачи с использованием методологий описания программных алгоритмов</p>	Модуль 1. Основные понятия, цели и методологии курса» Модуль 2 «Программные приложения	Устный опрос, задачи, тестирование	Зачет

ПК-3	Способен применять в профессиональной деятельности современные языки программирования и моделирования	<p>ПК-3.1 Использует объектно-ориентированную парадигму средств программирования и моделирования</p> <p>ПК-3.2 Демонстрирует навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p> <p>ПК-3.3 Демонстрирует навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разра-</p>	
------	-------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

<p><b>Знать:</b> объектно-ориентированную парадигму средств программирования и моделирования</p>			
<p><b>Знать:</b> принципы построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p>			
<p><b>Знать:</b> принципы алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки</p>			

		ботки					
--	--	-------	--	--	--	--	--

Код контролируемой компетенции	Формулировка контролируемой компетенции	Индикаторы достижения компетенции	Этап (уровень) освоения компетенции	Планируемые результаты обучения	Наименование модулей и (или) разделов дисциплины	Наименование оценочного средства	
						Текущий контроль	Промежуточная аттестация
ПК-1	Способность проводить описание прикладных процессов и информационного обеспечения решения прикладных задач	<p>ПК-1.1 Демонстрирует и использует знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов</p> <p>ПК-1.3 Демонстрирует навыки формализации прикладной задачи с использованием методологий описа-</p>	Второй этап (продвинутый уровень)	<p><b>Уметь:</b> демонстрировать и использовать знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов</p> <p><b>Уметь:</b> демонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов</p>	Основные понятия, цели и методологии курса» Модуль 2 «Программные приложения	Устный опрос, задачи, тестирование	Зачет

<p>ПК-3</p>	<p>Способен применять в профессиональной деятельности современные языки программирования и моделирования</p>	<p>ния программных алгоритмов</p> <p>ПК-3.1 Использует объектно-ориентированную парадигму средств программирования и моделирования</p> <p>ПК-3.2 Демонстрирует навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p> <p>ПК-3.3 Демонстрирует навыки алгоритмизации, разработки, отладки и те-</p>	
-------------	--------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

<p><b>Уметь:</b> использовать-объектно-ориентированную парадигму средств программирования и моделирования</p> <p><b>Уметь:</b> демонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p> <p><b>Уметь:</b> демонстрировать навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегриро-</p>			
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--

		стирования программ в различных интегрированных средах разработки		ванных средах разработки			
--	--	-------------------------------------------------------------------	--	--------------------------	--	--	--

Код контролируемой компетенции	Формулировка контролируемой компетенции	Индикаторы достижения компетенции	Этап (уровень) освоения компетенции	Планируемые результаты обучения	Наименование модулей и (или) разделов дисциплины	Наименование оценочного средства	
						Текущий контроль	Промежуточная аттестация
ПК-1	Способность проводить описание прикладных процессов и информационного обеспечения решения прикладных задач	<p>ПК-1.1 Демонстрирует и использует знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов</p> <p>ПК-1.3 Демонстрирует навыки формализации прикладной задачи с использованием методов</p>	Третий этап (высокий уровень)	<p><b>Владеть</b> знанием основных методологий описания архитектуры вычислительной системы и программных алгоритмов</p> <p><b>Владеть</b> навыками формализации прикладной задачи с использованием методов</p>	<i>Свободно владет способностью</i> использовать объектно-ориентированную парадигму средств программирования и моделирования тем систем	Устный опрос, задачи, тестирование	Зачет

<p>ПК-3</p>	<p>Способен применять в профессиональной деятельности современные языки программирования и моделирования</p>	<p>лизации прикладной задачи с использованием методологий описания программных алгоритмов</p> <p>ПК-3.1 Использует объектно-ориентированную парадигму средств программирования и моделирования</p> <p>ПК-3.2 Демонстрирует навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p>	
-------------	--------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--

<p>логий описания программных алгоритмов</p> <p><b>Владеть</b> объектно-ориентированной парадигмой средств программирования и моделирования</p> <p><b>Владеть</b> навыками построения, программирования и эксплуатации систем с использованием микропроцессорной техники;</p>			
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--	--	--

		ПК-3.3 Демонстрирует навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки		<b>Владеть</b> навыками алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки			
--	--	------------------------------------------------------------------------------------------------------------------------------------------	--	-----------------------------------------------------------------------------------------------------------------------------------	--	--	--

## 2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Компетенция	Планируемые результаты обучения, соотнесенные с индикаторами достижения компетенции (показатели достижения заданного уровня компетенции)	Уровни и критерии оценивания результатов обучения, шкалы оценивания			
		<i>Компетентность не сформирована</i>	<i>Пороговый уровень компетентности</i>	<i>Продвинутый уровень компетентности</i>	<i>Высокий уровень</i>
		<i>Неудовлетворительно Не зачтено</i>	<i>Удовлетворительно Зачтено</i>	<i>Хорошо Зачтено</i>	<i>Отлично Зачтено</i>
1	2	3	4	5	6

<b>ПК-1</b> Способность проводить описание прикладных процессов и информационного обеспечения решения прикладных задач	ПК-1.1 Демонстрирует и использует знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов	<i>Не способен</i> продемонстрировать и использовать знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов	<i>Частично способен</i> продемонстрировать и использовать знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов	<i>Владеет способностью</i> продемонстрировать и использовать знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов систем	<i>Свободно владеет способностью</i> продемонстрировать и использовать знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов
	<b>Знать:</b> основные методологии описания архитектуры вычислительной системы и программных алгоритмов	Допускает грубые ошибки при рассмотрении основных методологий описания архитектуры вычислительной системы и программных алгоритмов	Может изложить основы методологий описания архитектуры вычислительной системы и программных алгоритмов	Знает основы методологий описания архитектуры вычислительной системы и программных алгоритмов	Знает и объясняет основы методологий описания архитектуры вычислительной системы и программных алгоритмов

1	2	3	4	5	6
	<b>Уметь:</b> демонстрировать и использовать знание основных методологий описания архитектуры вычислительной системы и программных алгоритмов	Не умеет демонстрировать и использовать знание основных методологий описания ар-	Частично умеет демонстрировать и использовать знание основных методологий описа-	Способен в типовой ситуации демонстрировать и использовать знание основных ме-	Способен самостоятельно демонстрировать и использовать знание основных мето-

		хитектуры вычислительной системы и программных алгоритмов	ния архитектуры вычислительной системы и программных алгоритмов	тодологий описания архитектуры вычислительной системы и программных алгоритмов ;	логов описания архитектуры вычислительной системы и программных алгоритмов
	<b>Владеть:</b> знанием основных методологий описания архитектуры вычислительной системы и программных алгоритмов	Не владеет знанием основных методологий описания архитектуры вычислительной системы и программных алгоритмов	Частично владеет знанием основных методологий описания архитектуры вычислительной системы и программных алгоритмов	Владеет основами знаний основных методологий описания архитектуры вычислительной системы и программных алгоритмов	Свободно владеет основами знаний основных методологий описания архитектуры вычислительной системы и программных алгоритмов
	ПК-1.3 Демонстрирует навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	<i>Не способен</i> продемонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	<i>Частично способен</i> продемонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	<i>Владеет способностью</i> продемонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	<i>Свободно владеет способностью</i> продемонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов
	<b>Знать:</b> принципы формализации прикладной задачи с использованием методологий описания программных алгоритмов	Не знает принципы формализации прикладной задачи с использованием методологий опи-	Частично знает принципы формализации прикладной задачи с использованием ме-	Хорошо знает принципы формализации прикладной задачи с использованием ме-	Свободно владеет знанием о принципах формализации прикладной задачи с использованием

		сания программных алгоритмов	тодологий описа-ния программных алгоритмов	тодологий описа-ния программных алгоритмов	методологий опи-сания программных алгоритмов
	<b>Уметь:</b> демонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	Не умеет демонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	Частично умеет демонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	Хорошо умеет демонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов	Свободно умеет демонстрировать навыки формализации прикладной задачи с использованием методологий описания программных алгоритмов
	<b>Владеет:</b> навыками формализации прикладной задачи с использованием методологий описания программных алгоритмов	Не владеет навыками формализации прикладной задачи с использованием методологий описания программных алгоритмов	Частично владеет навыками формализации прикладной задачи с использованием методологий описания программных алгоритмов	Хорошо владеет навыками формализации прикладной задачи с использованием методологий описания программных алгоритмов	Свободно владеет навыками формализации прикладной задачи с использованием методологий описания программных алгоритмов
<b>ПК-3</b> Способен применять в профессиональной деятельности современные языки программирования и моделирования	ПК-3.1 Использует объектно-ориентированную парадигму средств программирования и моделирования	<i>Не способен</i> использовать объектно-ориентированную парадигму средств программирования и моделирования	<i>Частично способен</i> использовать объектно-ориентированную парадигму средств программирования и моделирования	<i>Владеет способностью</i> использовать объектно-ориентированную парадигму средств программирования и моделирования	<i>Свободно владеет способностью</i> использовать объектно-ориентированную парадигму средств программирования и моделирования тем систем

	<b>Знать:</b> объектно-ориентированную парадигму средств программирования и моделирования	Допускает грубые ошибки при инсталляции объектно-ориентированную парадигму средств программирования и моделирования	Может изложить основы объектно-ориентированной парадигмы средств программирования и моделирования	Знает основы объектно-ориентированной парадигмы средств программирования и моделирования	Знает и объясняет объектно-ориентированную парадигму средств программирования и моделирования
--	-------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------

1	2	3	4	5	6
	<b>Уметь:</b> применять объектно-ориентированную парадигму средств программирования и моделирования	Не умеет применить объектно-ориентированную парадигму средств программирования и моделирования	Частично анализирует, а также частично может применить объектно-ориентированную парадигму средств программирования и моделирования	Способен в типовой ситуации анализировать, синтезировать, применять объектно-ориентированную парадигму средств программирования и моделирования	Способен самостоятельно анализировать, синтезировать, применять объектно-ориентированную парадигму средств программирования и моделирования
	<b>Владеть:</b> навыками объектно-ориентированной парадигмы средств программирования и моделирования	Не владеет навыками объектно-ориентированной парадигмы средств программирования и моделирования	Частично владеет навыками объектно-ориентированной парадигмы средств программирования и моделирования	Владеет навыками объектно-ориентированной парадигмы средств программирования и моделирования	Свободно владеет навыками объектно-ориентированной парадигмы средств программирования и моделирования

	<b>ПК-3.2</b> Демонстрирует навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники;	<i>Не способен</i> продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники	<i>Частично способен</i> продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники	<i>Владеет способностью</i> продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники	<i>Свободно владеет способностью</i> продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники
	<b>Знать:</b> навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники;	Допускает грубые ошибки при построении, программировании и эксплуатации систем с использованием микропроцессорной техники;	Может изложить основы построения, программирования и эксплуатации систем с использованием микропроцессорной техники;	Знает основы построения, программирования и эксплуатации систем с использованием микропроцессорной техники;	Знает и объясняет основы построения, программирования и эксплуатации систем с использованием микропроцессорной техники;
	<b>Уметь:</b> продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники	Не умеет продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники	Частично умеет продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники	Хорошо умеет продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники	Свободно умеет продемонстрировать навыки построения, программирования и эксплуатации систем с использованием микропроцессорной техники
	<b>Владеет:</b> навыками построения, программирования и эксплуатации систем с использованием	Не владеет навыками построения, программирования	Частично владеет навыками построения, программирования	Хорошо владеет навыками построения, программирования	Свободно владеет навыками построения, программирования

	микропроцессорной техники	и эксплуатации систем с использованием микропроцессорной техники	вания и эксплуатации систем с использованием микропроцессорной техники	вания и эксплуатации систем с использованием микропроцессорной техники	вания и эксплуатации систем с использованием микропроцессорной техники
	<b>ПК-3.3</b> Демонстрирует навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	<b>Не способен</b> продемонстрировать навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	<b>Частично способен</b> продемонстрировать навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	<b>Владеет способностью</b> продемонстрировать навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	<b>Свободно владеет способностью</b> продемонстрировать навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки
	<b>Знать:</b> навыки алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки ;	Допускает грубые ошибки при алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	Может изложить основы алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	Знает основы алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	Знает и объясняет основы алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки
	<b>Уметь:</b> демонстрировать навыки алгоритмизации, разработки, отладки и тестирования программ в	Не умеет демонстрировать навыки алгоритмизации,	Частично умеет демонстрировать навыки алгоритми-	Хорошо умеет демонстрировать навыки алгоритми-	Свободно умеет демонстрировать навыки алгоритми-

	различных интегрированных средах разработки	разработки, отладки и тестирования программ в различных интегрированных средах разработки	зации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	зации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	зации, разработки, отладки и тестирования программ в различных интегрированных средах разработки
	<b>Владеет:</b> навыками алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	Не владеет навыками алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	Частично владеет навыками алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	Хорошо владеет навыками алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки	Свободно владеет навыками алгоритмизации, разработки, отладки и тестирования программ в различных интегрированных средах разработки

### **3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы**

#### **3.1. Первый этап (пороговой уровень)**

**ЗНАТЬ** (помнить и понимать): студент помнит, понимает и может продемонстрировать широкий спектр фактических, концептуальных, процедурных знаний.

##### **3.1.1. Перечень вопросов для определения входного рейтинга**

1. Виды программного обеспечения.
2. Предметная область. Модели предметной области.
3. Преобразование модели предметной области в схему базы данных.
4. Жизненный цикл программного обеспечения. Модели жизненного цикла.
5. Чем регламентируется ЖЦ ПО?
6. Какие группы процессов входят в состав ЖЦ ПО и какие процессы входят в состав каждой группы?
7. Какие процессы, по вашему мнению, наиболее часто используются в реальных процессах и почему?
8. Что понимается под стадией ЖЦ ПО и какие стадии входят в его состав?
9. Каково соотношение между стадиями и процессами ЖЦ ПО?
10. Какие процессы ЖЦ используются для получения достоверных оценок качества ПО?
11. Какие стадии ЖЦ чаще всего присутствуют в модели ЖЦ?
12. Каковы принципиальные особенности каскадной модели?
13. В чем заключаются преимущества и недостатки каскадной модели?
14. Каковы принципиальные особенности спиральной модели?
15. В чем заключаются преимущества и недостатки спиральной модели?

##### **3.1.2. Перечень вопросов к зачету**

1. Язык программирования Си. Общая характеристика языка: место в классификации языков, синтаксис и семантика языка, основные объекты языка, операторы, структура программы.
2. Понятие среды программирования. Компоненты среды. Компиляция и компоновка программы.
3. Синтаксические и логические ошибки. Тестирование и отладка программы.
4. Данные в языке Си: константы и переменные. Скалярные типы данных. Модификаторы типов.
5. Данные числовых типов в языке Си: объявление, характеристика, допустимые операции, приведение типов. Пример использования.
6. Операции языка Си. Приоритет операций. Оператор и операция присваивания в языке Си. Множественное присваивание. Выражения.
7. Функции форматного ввода и вывода. Параметры. Управляющая строка. Спецификаторы формата. Управляющие символы.
8. Алгоритмическая конструкция ветвления: полная и неполная форма, блок-схемы.  
Условная операция. Условный оператор в языке Си: структура оператора, полная и неполная формы, использование сложных условий. Пример на языке Си.
9. Алгоритмическая конструкция выбора: понятие, блок-схема. Оператор выбора в языке Си: структура оператора. Пример программы на Си.
10. Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла `for`: структура оператора, пример использования.

11. Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с предусловием (понятие, использование, блок-схема). Оператор цикла с предусловием `while` в языке Си: структура оператора, допустимые и недопустимые условия, пример использования.
12. Циклические алгоритмы: понятие, виды (перечислить). Алгоритмическая конструкция цикла с постусловием (понятие, использование, блок-схема). Оператор цикла с постусловием в языке Си: структура оператора, допустимые и недопустимые условия, пример использования.
13. Типовые циклические алгоритмы: максимум/минимум, сумма/произведение, количество.
14. Линейный массив: понятие массива, объявление, инициализация массива, индексация элементов. Формирование и вывод массива.
15. Типовые алгоритмы для работы с линейными массивами.
16. Задача сортировки массива. Алгоритм сортировки линейного массива методом "пузырька".
17. Двумерный массив. Объявление, инициализация двумерного массива, индексация элементов. Формирование и вывод двумерного массива.
18. Типовые алгоритмы для обработки двумерного массива (целиком).
19. Частичная обработка двумерного массива (по строкам или по столбцам). Типовые алгоритмы.
20. Указатели. Понятие указателя, объявление, инициализация. Операции, применимые к указателям. Указатель на указатель.
21. Использование указателей при работе с массивами и матрицами. Получение адресов и значений элементов. Последовательный перебор элементов. Примеры.
22. Функции в языке Си: понятие, объявление, прототипы функций. Вызов функции. Типы возвращаемых значений.
23. Функции в языке Си: параметры формальные и фактические, механизм передачи параметров. Передача параметров «по значению» и «по ссылке». Пример использования.
24. Локальные и глобальные переменные. Области действия и области видимости. Экранирование переменных.
25. Библиотечные функции. Заголовочные файлы. Подключение библиотек. Функции математической библиотеки.
26. Рекурсивное описание функций: база рекурсии, рекурсивный вызов, использование стека. Пример использования.
27. Динамическая память: выделение и освобождение памяти, размещение данных в динамической памяти. Выделение и освобождение памяти при работе с одиночными переменными и с массивами, изменение размера массива.
28. Динамическая память: выделение и освобождение памяти, размещение данных в динамической памяти. Выделение и освобождение памяти при работе с матрицами. Особенности обработки матриц при работе с динамической памятью.
29. Массив символов и строка в языке Си. Ввод и вывод строк. Простейшие алгоритмы сканирования и обработки строки. Пример.
30. Строка в языке Си: библиотечные функции для обработки строк. Примеры использования.

### 3.2. Второй этап (продвинутый уровень)

**УМЕТЬ** (применять, анализировать, оценивать, синтезировать): уметь использовать изученный материал в конкретных условиях и в новых ситуациях; осуществлять декомпозицию объекта на отдельные элементы и описывать то, как они соотносятся с целым, выявлять структуру объекта изучения; оценивать значение того или иного материала – научно-технической информации, исследовательских данных и т. д.; комбинировать

элементы так, чтобы получить целое, обладающее новизной

### 3.2.1. Тестовые задания

1. Что такое функция?
  - a) Некоторая часть программы, содержащая описание переменных и констант основной программы
  - b) Некоторая часть программы, имеющая собственное имя и которая может вызываться из основной программы**
  - c) Некоторая часть программы, содержащая вредоносный код, и блокирует определенные действия системы
  - d) Некоторая часть программы, в которой происходит начальная инициализация всех полей структур, массивов, переменных.
  
2. Что такое массив?
  - a) Именованный набор переменных имеющих различные типы данных, и располагающихся в одной памяти
  - b) Именованный набор переменных и функций, которые располагаются в одной области памяти
  - c) Именованный набор переменных имеющий один тип данных, и располагающихся в одной области памяти**
  - d) Именованный набор переменных имеющих символьный тип данных, и располагающихся в одной области памяти
  
3. Как написать следующее выражение на языке C «Переменной a присвоено значение b»?
  - a) `a==b`
  - b) `a=b`**
  - c) `b=a`
  - d) `a:=b`
  
4. Как написать следующее выражение «Второму элементу массива Myarray присвоено значение пяти»?
  - a) `int [1] Myarray=«пять»`
  - b) `int Myarray [1] = 5`**
  - c) `int Myarray [2] = «пять»`
  - d) `int Myarray [2] = 5`
  
5. Как написать следующее выражение «Если переменная index больше size то мы инкрементируем переменную count»?
  - a) **`if (index>size) { count++; }`**
  - b) `if (index<size) { count--; }`
  - c) `if (index>=size) { ++count; }`
  - d) `if (index<size) { --count; }`
  
6. Какой диапазон значений имеет тип int для 32-разрядных вычислительных систем:

- a) от 0 до 255
- b) от -32768 до 32767
- c) от 0 до 65535
- d) от 0 до 4 294 967 295**

7. Какой размер в байтах имеет переменная вещественного типа float

- a) 2
- b) 4**
- c) 8
- d) 10

8. Дан массив  $\text{int } L[3][3] = \{ \{ 2, 3, 4 \}, \{ 3, 4, 8 \}, \{ 1, 0, 9 \} \}$ ; Чему будет равно значение элемента этого массива  $L[1][2]$

- a) 2
- b) 3
- c) 4
- d) 8**

9. Объявление  $\text{char } *buf$ ; соответствует

- a) созданию символьной переменной buf
- b) созданию строковой переменной buf
- c) созданию указателя buf на символьное значение**
- d) созданию указателя buf на строку

10. Что называется прототипом функции?

- a) описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров
- b) описание функции, включая ее имя, тип возвращаемого значения, типы параметров
- c) имя функции и тип возвращаемого значения**
- d) описание функции, включая ее имя, тип возвращаемого значения, имена и типы параметров, тело функции

11. Как обозначается в языке C (C++) следующий режим работы с потоком - создание нового файла для записи и чтения?

- a) a+
- b) wb
- c) w+**
- d) w+b

12. Какая функция, описанная в заголовочном файле читает строку символов из файла?

- a) gets()
- b) fputs()
- c) fgets()**
- d) fscanf()

13. Какой размер массива M будет после выполнения кода:  
char M[ ]="\nGoodlive" ?

- a) 10
- b) 8
- c) 9**
- d) Не определен

14. В каких случаях необходимо использовать оператор return в теле функции?

- a) Всегда
- b) если необходимо, чтобы функция вернула значение**
- c) если необходимо обеспечить выход из функции в произвольном месте
- d) если указан тип возвращаемого значения, в том числе и void

15. При открытии файла выполняется следующее действие:

- a) физический файл связывается с логическим (файловой переменной)
- b) устанавливается тип файла (текстовый или бинарный)
- c) устанавливается вид (режим) использования файла**
- d) функцией открытия файла возвращается результат (ошибка)

16. Какое ключевое слово языка C++ используется для описания структурированного типа данных, все элементы которого в памяти начинаются с одного байта?

- a) struct
- b) union**
- c) enum
- d) template

17. Каким способом можно задать многострочный комментарий в языке C++

- a) /\*комментарии к программе\*/**
- b) //комментарии к программе//
- c) //комментарии к программе
- d) {комментарии к программе}

18. Логическое выражение может возвращать результат типа

- a) integer
- b) boolean**
- c) char
- d) logical

19. Выберите правильный вариант записи на языке C формулы  $0 < x < 10$

- a)  $x > 0, x \leq 10$
- b)  $0 < x \leq 10$
- c)  $x > 0 \text{ AND } x \leq 10$
- d)  $(x > 0) \text{ AND } (x < 10)$**

20. Укажите правильный вариант записи условного оператора в языке C

- a) IF x>0 Do y:=sqrt (x)
- b) IF y:=sqrt (x) then x>0
- c) IF x>0 then y:=sqrt (x)
- d) IF (x>0) { y:=sqrt (x)}**

21. Выберите правильный вариант записи на языке C следующего условия: « x принадлежит диапазону [0;10)»

- a) x>=0; x<10
- b) 0<=x<10
- c) (x>0 AND (x<=10))
- d) (x>=0) AND (x<10)**

22. Укажите группу, содержащую последовательность правильно записанных на языке C знаков операций отношений

- a) ~>, <=, ?
- b) =, <>, ><, >
- c) =, >=, <=, !=**
- d) ~ =>, = <, =, <

23. Тело какого цикла всегда будет выполнено хотя бы один раз, независимо от истинности условия:

- a) While
- b) Do While**
- c) For
- d) Нет такого цикла в языке C

24. В результате выполнения кода

```
int i=2;      switch (i)    { case 1: i += 2; case 2: i *= 3; case 6: i /= 2;
                default;  }
```

- a) переменная i примет значение 6
- b) переменная i примет значение 3**
- c) переменная i примет значение 2
- d) тело оператора switch не поменяет значение переменной i

25. Укажите директиву препроцессора, которую необходимо подключить для организации форматированного ввода-вывода данных:

- a) #include <iostream.h>
- b) #include <stdio.h>**
- c) #include <math.h>
- d) #include <conio.h>

### 3.2.2. Темы рефератов

1. История языков программирования.
2. Язык компьютера и человека.

3. Объектно-ориентированное программирование.
4. Непроцедурные системы программирования.
5. Искусственный интеллект и логическое программирование.
6. Языки манипулирования данными в реляционных моделях.
7. Макропрограммирование в среде Microsoft OFFICE.
8. «Визуальное» программирование. VISUAL BASIC, C, PROLOG.
9. Все о DELPHI.
10. Программирование на HTML, JAVA.
11. Издательская система TeX как система программирования.
12. Современные парадигмы программирования. Что дальше?
13. Никлаус Вирт. Структурное программирование. Pascal и Modula.
14. Что мы знаем о Fortran?
15. История языка Бейсик.
16. Язык Ассемблера.
17. Алгоритмический язык Ершова.
18. Все о Logo-мирах.
19. История программирования в лицах.
20. Язык программирования ADA.
21. Язык программирования PL/1.
22. Язык программирования Algol.
23. Язык программирования Си.
24. О фирмах-разработчиках систем программирования.
25. Языки программирования в СУБД.
26. О системах программирования для учебных целей.

### 3.3. Третий этап (высокий уровень)

**ВЛАДЕТЬ** навыками по применению теоретических и практических знаний и умений при решении ситуационных задач, практической направленности по дисциплине.

#### 3.3.1. Ситуационные задачи

**Задача 1.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается общее число оценок  $N$ , каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Класс> <Фамилия> <Инициалы> <Оценка> <Название предмета>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Для каждого учащегося определить количество оценок по каждому предмету (если по какому-либо предмету учащийся не получил ни одной оценки, то вывести для этого предмета число 0). Сведения о каждом учащемся выводить на отдельной строке, указывая класс, фамилию, инициалы и количество оценок по алгебре, геометрии и информатике. Данные располагать в порядке возрастания номера класса, а для одинаковых классов — в алфавитном порядке фамилий и инициалов.

**Задача 2.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается общее число оценок  $N$ , каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Название предмета> <Фамилия> <Инициалы> <Класс> <Оценка>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Вывести сведения об учащихся, имеющих по алгебре среднюю оценку не более 4.0: фамилию, инициалы, номер класса и среднюю оценку по алгебре (выводится с двумя дробными знаками). Учащихся, не имеющих ни одной оценки по алгебре, не учитывать. Сведения о каждом учащемся выводить на отдельной строке и располагать в алфавитном порядке их фамилий и инициалов. Если ни один из учащихся не удовлетворяет указанным условиям, то вывести текст «Требуемые учащиеся не найдены».

**Задача 3.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается общее число оценок  $N$ , каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Класс> <Фамилия> <Инициалы> <Название предмета> <Оценка>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Вывести сведения об учащихся, имеющих по информатике среднюю оценку не менее 4.0: номер класса, фамилию, инициалы и среднюю оценку по информатике (выводится с двумя дробными знаками). Сведения о каждом учащемся выводить на отдельной строке и располагать в порядке возрастания классов, а для одинаковых классов — в алфавитном порядке фамилий и инициалов. Если ни один из учащихся не удовлетворяет указанным условиям, то вывести текст «Требуемые учащиеся не найдены».

**Задача 4.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается название одного из предметов  $S$  и (через пробел) общее число оценок  $N$ . Каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Фамилия> <Инициалы> <Название предмета> <Оценка> <Класс>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Для каждого класса, присутствующего в наборе исходных данных, определить число учащихся, имеющих по предмету  $S$  среднюю оценку не более 3.5. Учащихся, не имеющих ни одной оценки по данному предмету, не учитывать. Сведения о каждом классе выводить на отдельной строке, указывая число найденных учащихся (число может быть равно 0) и номер класса. Данные упорядочивать по возрастанию числа учащихся, а для совпадающих чисел — по убыванию номера класса.

**Задача 5.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается название одного из предметов  $S$  и (через пробел) общее число оценок  $N$ . Каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Название предмета> <Фамилия> <Инициалы> <Оценка> <Класс>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапа-

зоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Для каждого класса, присутствующего в наборе исходных данных, определить число учащихся, имеющих по предмету  $S$  среднюю оценку не менее 3.5 и при этом не получивших ни одной двойки по этому предмету. Сведения о каждом классе выводить на отдельной строке, указывая номер класса и число найденных учащихся (число может быть равно 0). Данные упорядочивать по возрастанию номера класса.

**Задача 6.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается общее число оценок  $N$ , каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Класс> <Название предмета> <Фамилия> <Инициалы> <Оценка>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Найти всех *двоечников* — учащихся, получивших хотя бы одну двойку по какому-либо предмету. Вывести сведения о каждом из двоечников: номер класса, фамилию, инициалы и полученное число двоек. Сведения о каждом двоечнике выводить на отдельной строке и располагать по убыванию классов, а для одинаковых классов — в алфавитном порядке фамилий и инициалов. Если в наборе исходных данных нет ни одной двойки, то вывести текст «Требуемые учащиеся не найдены».

**Задача 7.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается общее число оценок  $N$ , каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Класс> <Оценка> <Фамилия> <Инициалы> <Название предмета>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Найти всех *хорошистов* — учащихся, не получивших ни одной двойки и тройки, но имеющих хотя бы одну четверку по какому-либо предмету. Вывести сведения о каждом хорошисте: полученное число четверок, фамилию, инициалы и номер класса. Сведения о каждом учащемся выводить на отдельной строке и располагать по возрастанию количества четверок, а при их равенстве — в алфавитном порядке фамилий и инициалов. Если в наборе исходных данных нет ни одного учащегося, удовлетворяющего указанным условиям, то вывести текст «Требуемые учащиеся не найдены».

**Задача 8.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается общее число оценок  $N$ , каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Класс> <Фамилия> <Инициалы> <Название предмета> <Оценка>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Для каждого класса найти *злостных двоечников* — учащихся, получивших в данном классе максимальное суммарное число двоек по всем предметам (число не должно быть нулевым). Вывести сведения о каждом из злостных двоечников: фами-

лию, инициалы, номер класса и полученное число двоек. Сведения о каждом двоечнике выводить на отдельной строке и располагать в алфавитном порядке их фамилий и инициа-

лов (сортировку по классам не проводить). Если в наборе исходных данных нет ни одной двойки, то вывести текст «Требуемые учащиеся не найдены».

**Задача 9.** На вход подаются сведения об оценках учащихся по трем предметам: алгебре, геометрии и информатике. В первой строке указывается общее число оценок  $N$ , каждая из последующих  $N$  строк содержит информацию об одной из оценок в формате

<Оценка> <Класс> <Фамилия> <Инициалы> <Название предмета>

Количество учащихся не превосходит 100, полных однофамильцев (с совпадающей фамилией и инициалами) среди учащихся нет. Класс описывается целым числом в диапазоне от 7 до 11, оценка — целым числом в диапазоне от 2 до 5. Название предмета указывается

с заглавной буквы. Для каждого класса найти *лучших учеников* — учащихся, получивших в данном классе максимальное суммарное число пятерок по всем предметам (число не должно быть нулевым). При поиске лучших учеников (в частности, при определении максимального суммарного числа пятерок) не следует учитывать учащихся, получивших хотя бы одну двойку или тройку. Вывести сведения о каждом из лучших учеников: номер класса, фамилию, инициалы и полученное число пятерок. Сведения о каждом учащемся выводить на отдельной строке и располагать по возрастанию классов, а для одинаковых классов — в алфавитном порядке фамилий и инициалов. Если в наборе исходных данных нет ни одного учащегося, удовлетворяющего указанным условиям, то вывести текст «Требуемые учащиеся не найдены».

**Задача 10.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200. Найти и вывести сумму цифр, входящих в исходный набор. Если набор не содержит цифр, то вывести слово «NO».

**Задача 11.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200. Найти сумму чисел, входящих в исходный набор (число образуют подряд идущие цифры; известно, что сумма чисел не превосходит 20000). Если исходный набор не содержит цифр, то вывести число  $-1$ .

**Задача 12.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200. Порядк идущие в наборе цифры образуют одно число; известно, что все числа имеют не более 4 разрядов (незначащие нули при подсчете количества разрядов не учитываются). Найти общую сумму всех однозначных и двузначных чисел и общее количество всех трехзначных и четырехзначных чисел. Вывести два полученных числа в одной строке в указанном порядке, разделяя их одним пробелом. Если набор не содержит чисел с требуемым количеством разрядов, то в качестве соответствующего числа вывести 0.

**Задача 13.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200. Составить из имеющихся в наборе нечетных цифр минимальное число и вывести строковое представление этого числа. Каждая цифра должна использоваться столько раз, сколько раз она встречается в исходном наборе. Если в наборе нет нечетных цифр, то вывести слово «NO».

**Задача 14.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200. Составить из имеющихся в наборе четных цифр максимальное число и вывести строковое представление этого числа. Каждая цифра должна использоваться столько раз, сколько раз она встречается в исходном наборе (за исключением ситуации, когда отсутствуют все четные цифры, кроме цифры 0; в этом случае надо вывести строку «0»). Если в наборе нет четных цифр, то вывести слово «NO».

**Задача 15.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200; в наборе обязательно присутствуют цифры. Составить из имеющихся цифр максимальное число, которое читается одинаково как слева направо, так и справа налево. Все цифры можно не использовать; незначащих нулей в числе не должно быть. Вывести строку, изображающую найденное число.

**Задача 16.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200. Составить из имеющихся в тексте нечетных цифр число максимальной длины, которое читается одинаково как слева направо, так и справа налево. Все цифры можно не использовать. Если требуемое число составить нельзя, то вывести слово «NO»; если можно, то в первой строке вывести слово «YES», а во второй — строку, изображающую найденное число. Если таких чисел несколько, то вывести минимальное из них.

**Задача 17.** На вход подается набор символов, оканчивающийся точкой. Количество символов не превышает 200. Составить из имеющихся в тексте четных цифр максимальное число, которое читается одинаково как слева направо, так и справа налево. Все цифры можно не использовать; незначащих нулей в числе не должно быть. Если требуемое число составить нельзя, то вывести слово «NO»; если можно, то в первой строке вывести слово «YES», а во второй — строку, изображающую найденное число.

**Задача 18.** На вход в первой строке подается целое положительное число  $N$ , затем следует текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содержится не более 80 символов. Зашифровать исходный текст и вывести его зашифрованный вариант. Шифрование заключается в замене каждой латинской буквы на букву, симметричную шифруемой букве относительно середины английского алфавита (например, «a» должна заменяться на «z», «b» на «y», «x» на «c»). Регистр букв при шифровании остается прежним. Если символ не является латинской буквой, то он не изменяется.

**Задача 19.** На вход в первой строке подаются два целых положительных числа:  $K$  и  $N$ , причем  $K$  не превосходит 25. Затем следует текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содержится не более 80 символов. Зашифровать исходный текст и вывести его зашифрованный вариант. Шифрование заключается в замене каждой латинской буквы на букву, расположенную в алфавите на  $K$ -й позиции после шифруемой буквы (алфавит считается *циклическим*: за буквой «z» опять следует буква «a»). Регистр букв при шифровании остается прежним. Если символ не является латинской буквой, то он не изменяется.

**Задача 20.** На вход в первой строке подается символ  $C$  — латинская буква, во второй строке — целое положительное число  $N$ . Затем следует зашифрованный текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содержится не более 80 символов. Известно, что шифрование выполнялось по тем же правилам, что и в задании ExamTaskC92, причем первый зашифрованный символ текста должен после расшифровки стать символом  $C$ . Расшифровать исходный текст и вывести его расшифрованный вариант. Первый зашифрованный символ не обязательно является первым символом текста (поскольку в начале текста может располагаться цифра или знак препинания), однако можно считать, что он всегда присутствует в первой строке текста.

**Задача 21.** На вход в первой строке подается целое положительное число  $N$ , затем следует текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содержится не более 80 символов. Зашифровать исходный текст и вывести его зашифрованный вариант. Шифрование заключается в следующем: каждая латинская буква в слове из  $L$  букв заменяется на букву, расположенную в алфавите на  $L$ -й позиции после шифруемой буквы (ал-

фавит считается циклическим: за буквой «z» опять следует буква «a»). Например, текст «A graу box.» должен быть заменен на «B kvес ега.». Регистр букв при шифровании оста-ется прежним. Словом считается последовательность латинских букв, не содержащая пробелов и знаков препинания. Если символ не является латинской буквой, то он не изме-няется.

**Задача 22.** На вход в первой строке подаются два целых положительных числа:  $K$  и  $N$ , причем  $K$  не превосходит 10. Затем следует текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содержится не более 80 символов. Зашифровать исход-ный текст и вывести его зашифрованный вариант. Шифрование заключается в следую-щем: каждая латинская буква в слове из  $L$  букв заменяется на букву, расположенную в ал-фавите на  $(K + L)$ -й позиции после шифруемой буквы (алфавит считается циклическим: за буквой «z» опять следует буква «a»). Например, текст «A graу box.» при  $K = 1$  должен быть заменен на «C lwfd fsb.». Регистр букв при шифровании остается прежним. Словом считается последовательность латинских букв, не содержащая пробелов и знаков препи-нания. Если символ не является латинской буквой, то он не изменяется.

**Задача 23.** На вход в первой строке подается символ  $C$  — латинская буква, во вто-рой строке — целое положительное число  $N$ . Затем следует зашифрованный текст на ан-глий-ском языке, состоящий из  $N$  строк. В каждой строке текста содержится не более 80 символов. Известно, что шифрование выполнялось по тем же правилам, что и в зада-нии ExamTaskC95, причем первый зашифрованный символ текста должен после расшиф-ровки стать символом  $C$ . Расшифровать исходный текст и вывести его расшифрованный вариант. Первый зашифрованный символ не обязательно является первым символом тек-ста (поскольку в начале текста может располагаться цифра или знак препинания), однако можно считать, что он всегда присутствует в первой строке текста.

**Задача 24.** На вход в первой строке подается целое положительное число  $N$ , затем следует текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содер-жится не более 80 символов. Вывести все символы, встретившиеся в исходном тексте, включая пробел и знаки препинания (без повторений). Символы выводить в одной строке (без раз-делителей) в порядке их первого появления в тексте.

**Задача 25.** На вход в первой строке подается целое положительное число  $N$ , затем следует текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содер-жится не более 80 символов. Вывести все символы, встретившиеся в исходном тексте, включая пробел и знаки препинания (без повторений). Символы выводить в одной строке (без раз-делителей) в порядке возрастания их кодов.

**Задача 26.** На вход в первой строке подается целое положительное число  $N$ , затем следует текст на английском языке, состоящий из  $N$  строк. В каждой строке текста содер-жится не более 80 символов. Вывести последовательность строчных латинских букв (a, b, c, ..., z) из исходного текста и число их повторений в этом тексте. Вывод производится в алфавит-ном порядке, каждая буква выводится на новой строке, после буквы указывается пробел и число ее повторений в тексте. Буквы, отсутствующие в тексте, не выводить.

## 3.5 Критерии оценивания контрольных заданий для использования в ФОС дисциплины

### 3.5.1. Критерии оценивания тестового задания:

Тестовые задания оцениваются по шкале: 1 балл за правильный от-вет, 0 баллов за неправильный ответ. Итоговая оценка по тесту формируется путем суммирования набранных баллов и отнесения их к общему количе-

ству вопросов в задании. Помножив полученное значение на 100%, можно привести итоговую оценку к традиционной следующим образом:

Процент правильных ответов	Оценка
90 – 100% <i>От 9 до 10 баллов и/или</i>	«отлично»
70 – 89 % <i>От 6 до 8 баллов и/или</i>	«хорошо»
51 – 69 % <i>От 3 до 5 баллов и/или</i>	«удовлетворительно»
менее 50 % <i>От 0 до 2 баллов и/или</i>	«неудовлетворительно»

### **3.5.2. Критерии оценивания реферата**

**(доклада):**

*От 4 до 5 баллов и/или «отлично»:* глубокое и хорошо аргументированное обоснование темы; четкая формулировка и понимание изучаемой проблемы; широкое и правильное использование относящейся к теме литературы и примененных аналитических методов; содержание исследования и ход защиты указывают на наличие навыков работы студента в данной области; оформление работы хорошее с наличием расширенной библиографии; защита реферата (или выступление с докладом) показала высокий уровень профессиональной подготовленности студента;

*От 2 до 3 баллов и/или «хорошо»:* аргументированное обоснование темы; четкая формулировка и понимание изучаемой проблемы; использование ограниченного, но достаточного для проведения исследования количества источников; работа основана на среднем по глубине анализе изучаемой проблемы и при этом сделано незначительное число обобщений; содержание исследования и ход защиты (или выступление с докладом) указывают на наличие практических навыков работы студента в данной области; реферат (или доклад) хорошо оформлен с наличием необходимой библиографии; ход защиты реферата (или выступления с докладом) показал достаточную профессиональную подготовку студента;

*От 1 до 2 баллов и/или «удовлетворительно»:* достаточное обоснование выбранной темы, но отсутствует глубокое понимание рассматриваемой проблемы; в библиографии преобладают ссылки на стандартные литературные источники; труды, необходимые для всестороннего изучения проблемы, использованы в ограниченном объеме; заметна нехватка компетентности студента в данной области знаний; оформление реферата (или доклада) содержит небрежности; защита реферата (или выступление с докладом) показала удовлетворительную профессиональную подготовку студента;

*0 баллов и/или «неудовлетворительно»:* тема реферата (или доклада) представлена в общем виде; ограниченное число использованных литературных источников; шаблонное изложение материала; суждения по исследуемой проблеме не всегда компетентны; неточности и неверные выводы по рассматриваемой литературе; оформление реферата (или доклада) с элементами заметных отступлений от общих требований; во время защиты (или выступления с докладом) студентом проявлена ограниченная профессиональная эрудиция.

### **3.5.3. Критерии оценивания на ситуационную задачу:**

*От 9 до 10 баллов и/или «отлично»:* студент глубоко и полно владеет методами решения задачи; решение выполнено оптимальным способом; по-

лученное решение соответствует условиям задачи; решение ситуационной задачи носит самостоятельный характер.

*От 6 до 8 баллов и/или «хорошо»:* решение студента соответствует указанным выше критериям, но в ход решения имеет отдельные неточности (несущественные ошибки); однако допущенные при решении ошибки исправляются самим студентом после дополнительных вопросов.

*От 3 до 5 баллов и/или «удовлетворительно»:* студент обнаруживает отсутствие навыков и понимание основных методик решения ситуационной задачи, но решение является неполным, имеет неточности и существенные ошибки; допущенные при решении ошибки не исправляются самим студентом после дополнительных вопросов.

*От 0 до 2 баллов и/или «неудовлетворительно»:* студент имеет разрозненные, бессистемные знания в области решаемой задачи; не владеет методами и подходами для решения задачи.

#### ***3.5.4. Критерии оценивания «Устный опрос»***

- оценка «зачтено» выставляется студенту, если обладает систематизированными знаниями, умениями и навыками по данному разделу дисциплины;

- оценка «не зачтено» выставляется студенту, если он не проявил систематизированных знаний, умений и навыков по данному разделу дисциплины.

#### ***3.5.5. Критерий оценивания на зачете***

Оценка на зачете определяется на основании следующих критериев:

- оценка «зачтено» ставится студенту, показавшему систематическое и достаточно глубокое знание учебного материала, умение свободно выполнять ситуационные и тестовые задания, предусмотренные программой, умение логически мыслить и формулировать свою позицию по проблемным вопросам. Зачет может получить студент, который правильно ответил на теоретические вопросы, допустив при этом недочеты не принципиального характера и правильно решившему предложенную на зачете задачу.

- оценка «не зачтено» ставится студенту, обнаружившему существенные пробелы в знании основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий.

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

Процедура оценки знаний умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций, производится преподавателем в форме текущего контроля и промежуточной аттестации.

Для повышения эффективности текущего контроля и последующей промежуточной аттестации студентов осуществляется структурирование дисциплины на модули. Каждый модуль учебной дисциплины включает в себя изучение законченного раздела, части дисциплины.

Основными видами текущего контроля знаний, умений и навыков в течение каждого модуля учебной дисциплины являются устный опрос, подготовка рефератов, решение ситуационных задач, тестирование.

Студент должен выполнить все контрольные мероприятия, предусмотренные в модуле учебной дисциплины к указанному сроку, после чего преподаватель проставляет балльные оценки, набранные студентом по результатам текущего контроля модуля учебной дисциплины.

Контрольное мероприятие считается выполненным, если за него студент получил оценку в баллах, не ниже минимальной оценки, установленной программой дисциплины по данному мероприятию.

Промежуточная аттестация обучающихся проводится в форме зачета.

Зачет проводится для оценки уровня усвоения обучающимся учебного материала лекционных курсов и лабораторно-практических занятий, а также самостоятельной работы. Оценка выставляется или по результатам учебной работы студента в течение семестра, или по итогам письменно-устного опроса, или тестирования на последнем занятии. Для дисциплин и видов учебной работы студента, по которым формой итогового отчета является зачет, определена оценка «зачтено», «не зачтено».

Оценка «зачтено» ставится в том случае, если обучающийся:

- владеет знаниями, выделенными в качестве требований к знаниям обучающихся в области изучаемой дисциплины;
- демонстрирует глубину понимания учебного материала с логическим и аргументированным его изложением;
- владеет основным понятийно-категориальным аппаратом по дисциплине;
- демонстрирует практические умения и навыки в области исследовательской деятельности.

Оценка «не зачтено» ставится в том случае, если обучающийся:

- демонстрирует знания по изучаемой дисциплине, но отсутствует глубокое понимание сущности учебного материала;
- допускает ошибки в изложении фактических данных по существу материала, представляется неполный их объем;
- демонстрирует недостаточную системность знаний;
- проявляет слабое знание понятийно-категориального аппарата по дисциплине;
- проявляет непрочность практических умений и навыков в области исследовательской деятельности.

В этом случае студент сдаёт зачёт в форме устных и письменных ответов на любые вопросы в пределах освоенной дисциплины.

Основным методом оценки знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций является

балльно-рейтинговая система, которая регламентируется Положением о балльно-рейтинговой системе оценки обучения в ФГБОУ ВО Белгородский ГАУ.

Основными видами поэтапного контроля результатов обучения студентов являются: рубежный рейтинг, творческий рейтинг, рейтинг личностных качеств, рейтинг сформированности прикладных практических требований, промежуточная аттестация.

Уровень развития компетенций оценивается с помощью рейтинговых баллов.

<b>Рейтинги</b>	<b>Характеристика рейтингов</b>	<b>Максимум баллов</b>
Рубежный	Отражает работу студента на протяжении всего периода изучения дисциплины. Определяется суммой баллов, которые студент получит по результатам изучения каждого модуля.	60
Творческий	Результат выполнения студентом индивидуального творческого задания различных уровней сложности, в том числе, участие в различных конференциях и конкурсах на протяжении всего курса изучения дисциплины.	5
Рейтинг личностных качеств	Оценка личностных качеств обучающихся, проявленных ими в процессе реализации дисциплины (модуля) (дисциплинированность, посещаемость учебных занятий, сдача вовремя контрольных мероприятий, ответственность, инициатива и др.)	10
Рейтинг сформированности прикладных практических требований	Оценка результата сформированности практических навыков по дисциплине (модулю), определяемый преподавателем перед началом проведения промежуточной аттестации и оценивается как «зачтено» или «не зачтено».	+
Промежуточная аттестация	Является результатом аттестации на окончательном этапе изучения дисциплины по итогам сдачи зачета или экзамена. Отражает уровень освоения информационно-теоретического компонента в целом и основ практической деятельности в частности.	25
Итоговый рейтинг	Определяется путём суммирования всех рейтингов	100

Общий рейтинг по дисциплине складывается из рубежного, творческого, рейтинга личностных качеств, рейтинга сформированности прикладных практических требований, промежуточной аттестации (экзамена или зачета).

Рубежный рейтинг – результат текущего контроля по каждому модулю дисциплины, проводимого с целью оценки уровня знаний, умений и навыков студента по результатам изучения модуля. Оптимальные формы и методы рубежного контроля: устные собеседования, письменные контрольные опросы, в т.ч. с использованием ПЭВМ и ТСО, результаты выполнения лабораторных и практических заданий. В качестве практических заданий могут выступать крупные части (этапы) курсовой работы или проекта, расчетно-графические задания, микропроекты и т.п.

Промежуточная аттестация – результат аттестации на окончательном этапе изучения дисциплины по итогам сдачи *зачета/ экзамена*, проводимого с целью проверки освоения информационно-теоретического компонента в целом и основ практической деятельности в частности. Оптимальные формы и методы выходного контроля: письменные экзаменационные или контрольные работы, индивидуальные собеседования.

Творческий рейтинг – составная часть общего рейтинга дисциплины, представляет собой результат выполнения студентом индивидуального творческого задания различных уровней сложности.

Рейтинг личностных качеств - оценка личностных качеств обучающихся, проявленных ими в процессе реализации дисциплины (модуля) (дисциплинированность, посещаемость учебных занятий, сдача вовремя контрольных мероприятий, ответственность, инициатива и др.

Рейтинг сформированности прикладных практических требований - оценка результата сформированности практических навыков по дисциплине (модулю), определяемый преподавателем перед началом проведения промежуточной аттестации и оценивается как «зачтено» или «не зачтено».

В рамках балльно-рейтинговой системы контроля успеваемости студентов, семестровая составляющая балльной оценки по дисциплине формируется при наборе заданной в программе дисциплины суммы баллов, получаемых студентом при текущем контроле в процессе освоения модулей учебной дисциплины в течение семестра.

Итоговая оценка /зачёта/ компетенций студента осуществляется путём автоматического перевода баллов общего рейтинга в стандартные оценки.

Максимальная сумма рейтинговых баллов по учебной дисциплине составляет 100 баллов.

Оценка «зачтено» ставится в том случае, если итоговый рейтинг студента составил 51 балл и более.

Оценка «не зачтено» ставится в том случае, если итоговый рейтинг студента составил менее 51 балла.

По дисциплине с экзаменом необходимо использовать следующую шкалу пересчета суммарного количества набранных баллов в четырехбалльную систему:

Неудовлетворительно	Удовлетворительно	Хорошо	Отлично
менее 51 балла	51-67 баллов	67,1-85 баллов	85,1-100 баллов

Издательство ФГБОУ ВО «Белгородский государственный аграрный  
университет имени В.Я. Горина

