

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»**

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.15 Технология программирования

Направление подготовки (специальность)
09.03.01 Информатика и вычислительная техника

Профиль подготовки (специализация)
“Автоматизированные системы обработки информации и управления”

Квалификация (степень) выпускника бакалавр

Форма обучения очная

1. Цели освоения дисциплины

Цель освоения дисциплины «Б1.В.15 Технология программирования» является:

- овладение студентами основными методами решения задач на компьютере на всех этапах процесса создания программного средства, языками программирования и их особенностей, основными принципами работы в системах программирования, основами программирования на языке высокого уровня.

2. Место дисциплины в структуре образовательной программы

Дисциплина «Б1.В.15 Технология программирования» относится к вариативной части. Требования к предшествующим знаниям представлены в таблице 2.1. Перечень дисциплин, для которых дисциплина «Б1.В.15 Технология программирования» является основополагающей, представлен в табл. 2.2.

Таблица 2.1 – Требования к пререквизитам дисциплины

Дисциплина	Раздел
Объектно-ориентированное программирование	Основные принципы объектно-ориентированного программирования

Таблица 2.2 – Требования к постреквизитам дисциплины

Дисциплина	Раздел
Алгоритмические языки и программирование	ПРИНЦИПЫ РАЗРАБОТКИ ПРОГРАММ

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Таблица 3.1 – Взаимосвязь планируемых результатов обучения по дисциплине и планируемых результатов освоения образовательной программы

Индекс и содержание компетенции	Знания	Умения	Навыки и (или) опыт деятельности
ОПК-1 способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	Этап 1: закономерности протекания информационных процессов в системах управления Этап2: принципы работы технических и программных средств	Этап 1: использовать методы и средства разработки алгоритмов и программ Этап 2: приемы структурного программирования	Этап 1: методами и средствами разработки документирования программ на языках высокого уровня для задач обработки числовой и символьной информации Этап 2: составление программ на языках высокого уровня для задач обработки числовой и символьной информации
ОПК-2 способностью осваивать методики использования программных средств для решения практических задач	Этап 1: технологию и методы производства программного продукта Этап 2: средства производства программного продукта	Этап 1: способы записи алгоритма на языке высокого уровня Этап 2: способы отладки, испытания и документирования программ	Этап 1: отладка программ на языках высокого уровня для задач обработки числовой и символьной информации Этап 2: тестирование и документирование программ на языках

			высокого уровня для задач обработки числовой и символьной информации
--	--	--	--

4. Объем дисциплины

Объем дисциплины «Б1.В.15 Технология программирования» составляет 4 зачетных единиц (144 академических часов). Распределение объема дисциплины на контактную работу обучающихся с преподавателем (КР) и на самостоятельную работу обучающихся (СР) по видам учебных занятий и по периодам обучения представлено в таблице 4.1.

Таблица 4.1 – Распределение объема дисциплины по видам учебных занятий и по периодам обучения, академические часы

№ п/п	Вид учебных занятий	Итого КР	Итого СР	Семестр № 4	
				КР	СР
1	2	3	4	5	6
1	Лекции (Л)	18		18	
2	Лабораторные работы (ЛР)				
3	Практические занятия (ПЗ)	36		36	
4	Семинары(С)				
5	Курсовое проектирование (КП)				
6	Рефераты (Р)				
7	Эссе (Э)				
8	Индивидуальные домашние задания (ИДЗ)				
9	Самостоятельное изучение вопросов (СИВ)		40		40
10	Подготовка к занятиям (ПкЗ)		40		40
11	Промежуточная аттестация	4	6	4	6
12	Наименование вида промежуточной аттестации			экзамен	
13	Всего:	58	86	58	86

5. Структура и содержание дисциплины

Структура дисциплины представлена в таблице 5.1.

Таблица 5.1 – Структура дисциплины

№ п/п	Наименования разделов и тем	Семестр	Объем работы по видам учебных занятий, академические часы											Коды формируемых компетенций
			лекции	лабораторная работа	практические занятия	семинары	курсовое проектирование	рефераты (эссе)	индивидуальные домашние задания	самостоятельное изучение вопросов	подготовка к занятиям	промежуточная аттестация	14	
1	2	3	4	5	6	7	8	9	10	11	12	13	14	
1.	Раздел 1 Технология программирования как технология разработки надежных программных средств	4	6		10					10	10			ОПК-1 ОПК-2
1.1.	Тема 1 Понятие о программном средстве	4	2		4					4	4			ОПК-1 ОПК-2
1.2.	Тема 2 Источники ошибок в программных средствах	4	2		4					4	4			ОПК-1 ОПК-2
1.3.	Тема 3 Специфика разработки программных средств	4	2		2					2	2			ОПК-1 ОПК-2
2.	Раздел 2 Описание программного средства	4	4		8					10	10			ОПК-1 ОПК-2
2.1.	Тема 4 Понятие внешнего описания	4	2		4					4	4			ОПК-1 ОПК-2
2.2.	Тема 5 Методы спецификации семантики функций	4	2		2					4	4			ОПК-1 ОПК-2
2.3.	Тема 6 Архитектура программного средства	4	-		2					2	2			ОПК-1 ОПК-2
3.	Раздел 3 Модульное программирование	4	4		8					10	10			ОПК-1 ОПК-2
3.1.	Тема 7 Разработка структуры программы	4	2		4					6	6			ОПК-1 ОПК-2
3.2.	Тема 8 Разработка программного модуля.)	4	2		4					4	4			ОПК-1 ОПК-2
4.	Раздел 4 Качество программного средства	4	4		10					10	10			ОПК-1 ОПК-2
4.1.	Тема 9	4	2		4					4	4			ОПК-1

№ п/п	Наименования разделов и тем	Семестр	Объем работы по видам учебных занятий, академические часы											Коды формируемых компетенций
			лекции	лабораторная работа	практические занятия	семинары	курсовое проектирование	рефераты (эссе)	индивидуальные домашние задания	самостоятельное изучение вопросов	подготовка к занятиям	промежуточная аттестация		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	
	Доказательство свойств программ													ОПК-2
4.2	Тема 10 Тестирование и отладка программного средства	4	2		2					2	2			ОПК-1 ОПК-2
4.3	Тема 11 Обеспечение функциональности и надежности программного средства	4	-		2					2	2			ОПК-1 ОПК-2
4.4	Тема 12 Обеспечение качества программного средства	4	-		2					2	2			ОПК-1 ОПК-2
5.	Контактная работа		18		36								4	
6.	Самостоятельная работа									40	40	6		
7.	Объем дисциплины в семестре		18		36					40	40	10		
8.	Всего по дисциплине		18		36					40	40	10		

5.2. Содержание дисциплины

5.2.1 – Темы лекций

№ п.п.	Наименование темы лекции	Объем, академические часы
Л-1	Понятие о программном средстве	2
Л-2	Источники ошибок в программных средствах	2
Л-3	Специфика разработки программных средств	2
Л-4	Понятие внешнего описания	2
Л-5	Методы спецификации семантики функций	2
Л-6	Архитектура программного средства	-
Л-7	Разработка структуры программы	2
Л-8	Разработка программного модуля	2
Л-9	Доказательство свойств программ	2
Л-10	Тестирование и отладка программного средства	2
Л-11	Обеспечение функциональности и надежности программного средства	-
Л-12	Обеспечение качества программного средства	-
Итого по дисциплине		18

5.2.2 – Темы лабораторных работ (не предусмотрено учебным планом)

5.2.3 – Темы практических занятий

№ п.п.	Наименование темы лабораторной работы	Объем, академические часы
ПР-1	Понятие о программном средстве	4
ПР-2	Источники ошибок в программных средствах	4
ПР-3	Специфика разработки программных средств	2
ПР-4	Понятие внешнего описания	4
ПР-5	Методы спецификации семантики функций	2
ПР-6	Архитектура программного средства	2
ПР-7	Разработка структуры программы	4
ПР-8	Разработка программного модуля	4
ПР-9	Доказательство свойств программ	4
ПР-10	Тестирование и отладка программного средства	2
ПР-11	Обеспечение функциональности и надежности программного средства	2
ПР-12	Обеспечение качества программного средства	2
Итого по дисциплине		36

5.2.4 – Темы семинарских занятий (не предусмотрены учебным планом)

5.2.5 Темы курсовых работ (проектов) (не предусмотрены учебным планом)

5.2.6 Темы рефератов (не предусмотрены)

5.2.7 Темы эссе (не предусмотрены)

5.2.8 Темы индивидуальных домашних заданий (не предусмотрены)

5.2.9 – Вопросы для самостоятельного изучения

№ п.п.	Наименование темы	Наименование вопросов	Объем, академические часы
1	Технология программирования как технология разработки надежных программных средств	Обеспечение контроля правильности принимаемых решений	10
2	Описание программного средства	Слоистая архитектура программного средства	10
3	Модульное программирование	Цель разработки структуры программы	10
4	Качество программного средства	Завершимость выполнения программы	10
Итого по дисциплине			40

6. Учебно-методическое и информационное обеспечение дисциплины

6.1. Основная литература, необходимая для освоения дисциплины

1. Тарасов А. Д., Боровский А. С., Баловнева Е. Г. Основы программирования на языке Паскаль: учебное пособие. – Оренбург: Издательский центр ОГАУ, 2006.
2. Костюк Ю. Л. Основы разработки алгоритмов: учебное пособие [электронный ресурс] / Ю. Л. Костюк, И. Л. Фукс. – М.: БИНОМ. Лаборатория знаний, 2010. – 286 с.: ил. ЭБС “Книгафонд”

6.2. Дополнительная литература, необходимая для освоения дисциплины

3. Тарасов А. Д., Матвеев А. Г. Методические рекомендации к летней практике по программированию / А. Д. Тарасов, А. Г. Матвеев. – Оренбург: Издательский центр ОГАУ, 2010. – 26 с.
4. Меняев М. Ф. Информатика и основы программирования: учебное пособие / М. Ф. Меняев. - М.: Омега-Л, 2007. - 458 с.
5. Рапаков Г. Г., Ржеуцкая С. Ю. Программирование на языке Pascal. [электронный ресурс] - СПб.: БХВ-Петербург, 2004. – 480 с.: ил. ЭБС “Книгафонд”

6.3. Методические указания для обучающихся по освоению дисциплины и другие материалы к занятиям

Электронное учебное пособие включающее:
- конспект лекций;
- методические указания по выполнению практических работ.

6.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

Электронное учебное пособие включающее:
- методические рекомендации по самостоятельному изучению вопросов;
- методические рекомендации по подготовке к занятиям.

6.5 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

Lazarus

6.6 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. <http://www.knigafund.ru/> - ЭБС
2. <http://e.lanbook.com/> - ЭБС
3. <http://rucont.ru/> - ЭБС
4. <http://elibrary.ru/defaultx.asp> - ЭБС
5. <http://www.rsl.ru> Российская государственная библиотека (РГБ)
6. <http://www.edu.ru/> - федеральный портал российского образования

7. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Занятия лекционного типа проводятся в аудитории, оборудованной мультимедиапроектором, компьютером, учебной доской.

Таблица 7.1 – Материально-техническое обеспечение практических занятий

№ п.п.	Наименование темы	Название специализированной лаборатории	Название спецоборудования	Название технических и электронных средств обучения и контроля знаний
ПР-1	Понятие о программном средстве	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-2	Источники ошибок в программных средствах	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-3	Специфика разработки программных средств	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-4	Понятие внешнего описания	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-5	Методы спецификации семантики функций	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-6	Архитектура программного средства	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-7	Разработка структуры	947 – Лаборатория сетевых и	ПЭВМ (по	Lazarus

	программы	мультимедийных технологий 953 – Лаборатория интеллектуальных систем	количеству обучающихся)	
ПР-8	Разработка программного модуля	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-9	Доказательство свойств программ	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-10	Тестирование и отладка программного средства	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-11	Обеспечение функциональности и надежности программного средства	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus
ПР-12	Обеспечение качества программного средства	947 – Лаборатория сетевых и мультимедийных технологий 953 – Лаборатория интеллектуальных систем	ПЭВМ (по количеству обучающихся)	Lazarus

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине представлен в Приложении 1.

Программа разработана в соответствии с Федеральным государственным образовательным стандартом высшего образования по направлению подготовки 09.03.01 Информатика и вычислительная техника, утвержденным приказом Министерства образования и науки РФ от 12 января 2016 г. № 5.

Разработал(и): _____

М.А. Корякина

**МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ОРЕНБУРГСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ»**

Приложение

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ
ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ
АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ
Б1.В.15 Технология программирования**

**Направление подготовки (специальность)
09.03.01 Информатика и вычислительная техника**

**Профиль подготовки (специализация)
“Автоматизированные системы обработки информации и управления”**

Квалификация (степень) выпускника бакалавр

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

Наименование и содержание компетенции

ОПК-1 способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем

Знать:

Этап 1: закономерности протекания информационных процессов в системах управления
Этап2: принципы работы технических и программных средств

Уметь:

Этап 1: использовать методы и средства разработки алгоритмов и программ
Этап 2: приемы структурного программирования

Владеть:

Этап 1: методами и средствами разработки документирования программ на языках высокого уровня для задач обработки числовой и символьной информации
Этап 2: составление программ на языках высокого уровня для задач обработки числовой и символьной информации

Наименование и содержание компетенции

ОПК-2 способностью осваивать методики использования программных средств для решения практических задач

Знать:

Этап 1: технологию и методы производства программного продукта
Этап 2: средства производства программного продукта

Уметь:

Этап 1: способы записи алгоритма на языке высокого уровня
Этап 2: способы отладки, испытания и документирования программ

Владеть:

Этап 1: отладка программ на языках высокого уровня для задач обработки числовой и символьной информации
Этап 2: тестирование и документирование программ на языках высокого уровня для задач обработки числовой и символьной информации

1. Показатели и критерии оценивания компетенций на различных этапах их формирования

Таблица 1 - Показатели и критерии оценивания компетенций на 1 этапе

Наименование компетенции	Критерии сформированности компетенции	Показатели	Способы оценки
1	2	3	4
ОПК-1 способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	Владеть способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	Знать: методами и средствами разработки документирования программ на языках высокого уровня для задач обработки числовой и символьной информации	индивидуальный устный опрос, практическое решение задач, тестирование.

		<p>Уметь: использовать методы и средства разработки алгоритмов и программ</p> <p>Владеть: закономерности протекания информационных процессов в системах управления</p>	
ОПК-2 способностью осваивать методики использования программных средств для решения практических задач	Владеть способностью осваивать методики использования программных средств для решения практических задач	<p>Знать: технологию и методы производства программного продукта</p> <p>Уметь: способы записи алгоритма на языке высокого уровня</p> <p>Владеть: отладка программ на языках высокого уровня для задач обработки числовой и символьной информации</p>	индивидуальный устный опрос, практическое решение задач, тестирование.

Таблица 2 - Показатели и критерии оценивания компетенций на 2 этапе

Наименование компетенции	Критерии сформированности компетенции	Показатели	Способы оценки
1	2	3	4
ОПК-1 способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	Владеть способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем	<p>Знать: принципы работы технических и программных средств</p> <p>Уметь: приемы структурного программирования</p> <p>Владеть: составление программ на языках высокого уровня для задач обработки числовой и символьной информации</p>	индивидуальный устный опрос, практическое решение задач, тестирование.
ОПК-2 способностью осваивать методики использования программных средств для решения	Владеть способностью осваивать методики использования программных средств для решения	<p>Знать: тестирование и документирование программ на языках высокого уровня для задач обработки числовой и</p>	индивидуальный устный опрос, практическое решение задач, тестирование.

практических задач	практических задач	символьной информации Уметь: способы отладки, испытания и документирования программ Владеть: средства производства программного продукта	
--------------------	--------------------	--	--

2. Шкала оценивания

Университет использует систему оценок соответствующего государственным регламентам в сфере образования и позволяющую обеспечивать интеграцию в международное образовательное пространство. Система оценок и описание систем оценок представлены в таблицах 3 и 4.

Таблица 3 - Система оценок

Диапазон оценки, в баллах	Экзамен		Зачет
	европейская шкала (ECTS)	традиционная шкала	
[95;100]	A – (5+)	отлично – (5) хорошо – (4) удовлетворительно – (3) неудовлетворительно – (2)	зачтено незачтено
[85;95)	B – (5)		
[70,85)	C – (4)		
[60;70)	D – (3+)		
[50;60)	E – (3)		
[33,3;50)	FX – (2+)		
[0;33,3)	F – (2)		

Таблица 4 - Описание системы оценок

ECTS	Описание оценок	Традиционная шкала
A	Превосходно – теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному.	
B	Отлично – теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному.	отлично (зачтено)

C	Хорошо – теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено максимальным числом баллов, некоторые виды заданий выполнены с ошибками.	хорошо (зачтено)
D	Удовлетворительно – теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки.	удовлетворительно (зачтено)
E	Посредственно – теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному	удовлетворительно (незачтено)
FX	Условно неудовлетворительно – теоретическое содержание курса освоено частично, необходимые практические навыки работы не сформированы, большинство предусмотренных программой обучения учебных заданий не выполнено, либо качество их выполнения оценено числом баллов, близким к минимальному; при дополнительной самостоятельной работе над материалом курса возможно повышение качества выполнения учебных заданий.	неудовлетворительно (незачтено)
F	Безусловно неудовлетворительно – теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, все выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к какому-либо значимому повышению качества выполнения учебных заданий.	неудовлетворительно (незачтено)

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

Таблица 5 – ОПК-1 способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем. Этап 1

<p>Наименование знаний, умений, навыков и (или) опыта деятельности</p>	<p>Формулировка типового контрольного задания или иного материала, необходимого для оценки знаний, умений, навыков и (или) опыта деятельности</p>
<p>Знать: методами и средствами разработки документирования программ на языках высокого уровня для задач обработки числовой и символьной информации</p>	<p>1. Тип функции определяется: 1) типом ее аргументов 2) использованием в программе 3) типом ее описания 4) типом возвращаемого ею значения Ответ: 4) типом возвращаемого ею значения</p> <p>2. Автоматические объекты: 1) существуют во время выполнения блока и теряют свои значения при выходе из него 2) хранятся вне любой функции и существуют в течение выполнения всей программы 3) являются объектами статического класса памяти 4) можно инициализировать только выражениями с константами и с указателями на ранее описанные объекты Ответ: 1) существуют во время выполнения блока и теряют свои значения при выходе из него</p> <p>3. Макропрограмма должна состоять: 1) из списка макросов 2) из списка макропеременных 3) из списка макроимен 4) из макроимени и заключенного в круглые скобки списка аргументов Ответ: 4) из макроимени и заключенного в круглые скобки списка аргументов</p> <p>4. Альтернатива – это: 1) композиция разных действий 2) вариант 3) конструкция ветвления 4) шаг выполнения программы Ответ: 3) конструкция ветвления</p> <p>5. Итерация – это: 1) шаг выполнения программы 2) циклическая конструкция алгоритма 3) язык программирования 4) функция прерывания Ответ: 1) шаг выполнения программы</p>
<p>Уметь: использовать методы и средства разработки алгоритмов и программ</p>	<p>1. Вычислительные процессы, используемые для решения различного рода задач на ЭВМ, в общем виде могут быть разделены на: 1) линейные, разветвляющиеся и циклические 2) высокого и низкого уровня 3) машинно-зависимые и машинно-независимые 4) словесные, словесно формульные и графические Ответ: 1) линейные, разветвляющиеся и циклические</p> <p>2. Языки высокого уровня делятся на: 1) процедурные, логические и объектно-ориентированные;</p>

	<p>2) машинно-зависимые и машинно-независимые;</p> <p>3) проблемно-ориентированные и универсальные;</p> <p>4) алгоритмические и неалгоритмические.</p> <p>Ответ: 1) процедурные, логические и объектно-ориентированные;</p> <p>3. Переменная в программировании полностью характеризуется:</p> <ol style="list-style-type: none"> 1) именем 2) именем, значением и типом 3) именем и типом 4) именем и значением 5) значением 4) ряд <p>Ответ: 2) именем, значением и типом</p>
Владеть: закономерности протекания информационных процессов в системах управления	<p>1. В языке С++ лексема – это:</p> <ol style="list-style-type: none"> 1) набор специальных символов и директив 2) множество строк, определяющих состояние программы 3) процедура, выполняющая определенные задания 4) последовательности символом языка, разделяющиеся пробелами и другими неграфическими символами <p>Ответ: 4) последовательности символом языка, разделяющиеся пробелами и другими неграфическими символами</p> <p>2. В языке С++ указатель – это:</p> <ol style="list-style-type: none"> 1) специальный значок, показывающий, что это динамическая переменная 2) символическое представление адреса ячейки памяти 3) переменная, которая содержит адрес объекта в памяти компьютера 4) метка в виде идентификатора с двоеточием <p>Ответ: 2) символическое представление адреса ячейки памяти</p> <p>3) переменная, которая содержит адрес объекта в памяти компьютера</p>

Таблица 6 - ОПК-1 способностью инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем. Этап 2

Наименование знаний, умений, навыков и (или) опыта деятельности	Формулировка типового контрольного задания или иного материала, необходимого для оценки знаний, умений, навыков и (или) опыта деятельности
Знать: принципы работы технических и программных средств	<p>1. Дедуктивный принцип – это:</p> <ol style="list-style-type: none"> 1) когда определяется связь между входными, выходными данными и процессами обработки 2) принцип построения модели от частного к общему 3) упаковывание информации и абстрактных типов данных 4) принцип построения модели от общего к частному <p>Ответ: 4) принцип построения модели от общего к частному</p> <p>2. Индуктивный принцип – это:</p> <ol style="list-style-type: none"> 1) когда определяется связь между входными, выходными данными и процессами обработки 2) принцип построения модели от частного к общему 3) упаковывание информации и абстрактных типов данных 4) принцип построения модели от общего к частному <p>Ответ: 2) принцип построения модели от частного к общему</p> <p>3. Линейный связный список – это:</p>

	<p>1) конечный набор пар, состоящих из информативных и указующих частей 2) рекурсивная конструкция алгоритма 3) совокупность динамических переменных 4) массив указателей</p> <p>Ответ: 1) конечный набор пар, состоящих из информативных и указующих частей</p> <p>4. Сортировка – это:</p> <p>1) процесс нахождения в заданном множестве объекта 2) процесс перегруппировки заданного множества объектов в некотором порядке 3) установка индексов элементов в возрастающем порядке 4) обработка элементов в алфавитном порядке</p> <p>Ответ: 2) процесс перегруппировки заданного множества объектов в некотором порядке</p> <p>5. Композиция – это:</p> <p>1) циклическая конструкция алгоритма, состоящая из многократного повторения одного блока действий 2) линейная конструкция алгоритма, состоящая из последовательно следующих друг за другом функциональных вершин 3) конструкция ветвления, имеющая предикатную вершину 4) механизм языка, позволяющий описать новый класс на основе существующего (родительского) класса</p> <p>Ответ: 2) линейная конструкция алгоритма, состоящая из последовательно следующих друг за другом функциональных вершин</p>
Уметь: приемы структурного программирования	<p>1. Фактический адрес в указателях – это:</p> <p>1) строка 2) указатель 3) число 4) буква</p> <p>Ответ: 3) число</p> <p>2. Составной оператор – это:</p> <p>1) последовательность операторов, заключенная в фигурные скобки { } 2) последовательность операторов, заключенная в квадратные скобки [] 3) последовательность операторов, заключенная в операторные скобки begin ... end 4) последовательность операторов, заключенная в круглые скобки ()</p> <p>Ответ: 1) последовательность операторов, заключенная в фигурные скобки { }</p> <p>3. Спецификация типа – это:</p> <p>1) задание типа переменной 2) список переменных 3) перечисление всех переменных, которые использовались в программе 4) список типов переменных, которые использовались в программе</p> <p>Ответ: 1) задание типа переменной</p>
Владеть: составление программ на языках высокого уровня для задач обработки числовой и символьной информации	<p>1. Битовая операция инверсии битов обозначается:</p> <p>1) \~ 2) ~ 3) >> 4) <<</p> <p>Ответ: 1) \~</p> <p>2. Битовая операция исключающего «или» обозначается:</p> <p>1) \~ 2) ~ 3) 4) &&</p> <p>Ответ: +2) ~</p>

Таблица 7 – ОПК-2 способностью осваивать методики использования программных средств для решения практических задач. Этап 1

Наименование	Формулировка типового контрольного задания или иного
--------------	--

знаний, умений, навыков и (или) опыта деятельности	материала, необходимого для оценки знаний, умений, навыков и (или) опыта деятельности
Знать: технологию и методы производства программного продукта	<p>1. Тестирование программы – это:</p> <ol style="list-style-type: none"> 1) оценивание ресурсов компьютера, на котором будет работать программа 2) перевод проекта в форму программы для конкретного компьютера 3) системный подход к построению алгоритма с использованием декомпозиции и синтеза 4) процесс исполнения программы с целью выявления ошибок <p>Ответ: 4) процесс исполнения программы с целью выявления ошибок</p> <p>2. Инспекция при тестировании – это:</p> <ol style="list-style-type: none"> 1) надзор за изменением состояний переменных 2) отслеживание логических ошибок 3) набор процедур и приемов обнаружения ошибок 4) надзор за соответствием типов и атрибутов переменных <p>Ответ: 3) набор процедур и приемов обнаружения ошибок</p> <p>3. Границные условия в тестах – это:</p> <ol style="list-style-type: none"> 1) ситуации, возникающие на, выше или ниже границ входных и выходных классов эквивалентности 2) тестовые задания, имеющие наивысшую вероятность обнаружения ошибок 3) выход индексов заданий за пределы допустимых 4) начальные и конечные условия границы применимости теста <p>Ответ: 1) ситуации, возникающие на, выше или ниже границ входных и выходных классов эквивалентности</p> <p>4. Если данные размещены на внешнем носителе, то доступ к ним возможен:</p> <ol style="list-style-type: none"> 1) моментальный 2) прямой 3) последовательный 4) выборочный <p>Ответ: 3) последовательный</p> <p>5. Если данные размещены в оперативной памяти, то доступ к ним возможен:</p> <ol style="list-style-type: none"> 1) прямой 2) параллельный 3) последовательный 4) перебором <p>Ответ: 1) прямой</p>
Уметь: способы записи алгоритма на языке высокого уровня	<p>1. Чтобы подключить заголовочный файл в программу на C++, например iostream необходимо написать:</p> <ol style="list-style-type: none"> 1) include #iostream,h; 2) #include <> с iostream внутри скобок 3) #include <>; с iostream.h внутри скобок 4) include (iostreamh) <p>Ответ: 2) #include <> с iostream внутри скобок</p> <p>2. Название C++ предложил:</p> <ol style="list-style-type: none"> 1) Дональд Кнут 2) Рик Масситти 3) Бьорн Страуструп 4) Кэн Томпсон <p>Ответ: 3) Бьорн Страуструп</p> <p>3. Какая из следующих записей - правильный комментарий в C++?</p> <ol style="list-style-type: none"> 1) {комментарий} 2) /* Комментарии */ 3) ** Комментарий ** +4) /* комментарий */ <p>Ответ: +4) /* комментарий */</p>
Владеть: отладка	1. Строковым литералом будет:

<p>программ на языках высокого уровня для задач обработки числовой и символьной информации</p>	<p>1) «sq» 2) %q 3) «s» 4) «qsqs» Ответ: 4) «qsqs»</p> <p>2. Оператор INT в C++ применяется для: 1) переопределения диапазона целых чисел 2) преобразования переменной к целому типу 3) описания переменных целого типа 4) прибавления единицы к коду символа Ответ: 2) преобразования переменной к целому типу</p>
--	--

Таблица 8 – ОПК-2 способностью осваивать методики использования программных средств для решения практических задач. Этап 2

<p>Наименование знаний, умений, навыков и (или) опыта деятельности</p>	<p>Формулировка типового контрольного задания или иного материала, необходимого для оценки знаний, умений, навыков и (или) опыта деятельности</p>
<p>Знать: тестирование и документирование программ на языках высокого уровня для задач обработки числовой и символьной информации</p>	<p>1. Процедура линейного поиска – это: 1) просмотр массива с конца 2) просмотр массива с середины 3) сравнение эталона осуществляется с элементом, расположенным в середине массива 4) последовательный просмотр всех элементов массива и сравнение их с эталоном Ответ: 4) последовательный просмотр всех элементов массива и сравнение их с эталоном</p> <p>2. Процедура поиска делением пополам заключается: 1) в просмотре массива с конца до середины 2) в просмотре массива с середины 3) в сравнении эталона с элементом, расположенным в середине массива 4) в последовательном просмотре всех элементов массива и сравнении их с эталоном Ответ: 3) в сравнении эталона с элементом, расположенным в середине массива</p> <p>3. Дан алгоритм сортировки: определяется минимальный элемент среди всех и меняется местами с первым и т. д., начиная со второго. Вид сортировки: 1) метод прямого включения 2) метод прямого выбора 3) пузырьковый метод 4) с помощью «дерева» Ответ: 2) метод прямого выбора</p> <p>4. Деструктивность процесса тестирования проявляется в следующем: 1) тест удачный, если обнаружена ошибка 2) тест удачный, если проведен без ошибок 3) тест неудачный, если обнаружена еще не выявленная ошибка 4) тест неудачный, если все задания некорректны Ответ: 1) тест удачный, если обнаружена ошибка</p> <p>5. Тестирование программы как черного ящика заключается в следующем: 1) знаем, какие данные будут на выходе 2) не знаем, какие данные подаем на входе</p>

	<p>3) анализ входных данных и результатов работы программы 4) управляем логикой программы, используя ее внутреннюю структуру</p> <p>Ответ: 3) анализ входных данных и результатов работы программы</p>
Уметь: способы отладки, испытания и документирования программ	<p>1. Формальный аргумент – это: 1) конкретное значение, присвоенное этой переменной вызывающей программой 2) переменная в вызываемой программе 3) строка, которая пишется в скобках функции 4) строка, которая пишется в скобках процедуры</p> <p>Ответ: 2) переменная в вызываемой программе</p> <p>2. Фактический аргумент – это: 1) конкретное значение, присвоенное этой переменной вызывающей программой 2) переменная в вызываемой программе 3) строка, которая пишется в скобках функции 4) строка, которая пишется в скобках процедуры</p> <p>Ответ: 1) конкретное значение, присвоенное этой переменной вызывающей программой</p> <p>3. Писать #include <stdio.h> нужно для: 1) подключения файла, содержащего макроопределения для работы функций из стандартной библиотеки ввода-вывода 2) позволяет дать в программе макроопределения (или задать макросы) 3) переопределения не только константы, но и целых программных конструкций 4) замены каждого параметра в строке лексем на соответствующий аргумент макровызова</p> <p>Ответ: 1) подключения файла, содержащего макроопределения для работы функций из стандартной библиотеки ввода-вывода</p>
Владеть: средства производства программного продукта	<p>1. Обращение к функции форматного ввода имеет вид: +1) scanf(<формат>, <&имя1>, <&имя2>, ..., <&имяN>) 2) printf (<формат>, <&имя1>, <&имя2>, ..., <&имяN>) 3) scanf(<формат>, <имя1>, <имя2>, ..., <имяN>) 4) printf (<формат>, <имя1>, <имя2>, ..., <имяN>)</p> <p>Ответ: +1) scanf(<формат>, <&имя1>, <&имя2>, ..., <&имяN>)</p> <p>2. Идентификатором будет: +1) schetchik get_line a12 Param1_ab 2) %ab 12abc -x schetchik 3) \b ab 12abc -x schetchik 4) * ab 12abc -x schetchik</p> <p>Ответ: +1) schetchik get_line a12 Param1_ab</p>

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

В процессе изучения дисциплины предусмотрены следующие формы контроля: текущий, промежуточный контроль (зачет и экзамен), контроль самостоятельной работы студентов.

Текущий контроль успеваемости обучающихся осуществляется по всем видам контактной и самостоятельной работы, предусмотренным рабочей программой дисциплины. Текущий контроль успеваемости осуществляется преподавателем, ведущим аудиторные занятия.

Текущий контроль успеваемости может проводиться в следующих формах:

- устная (устный опрос, защита письменной работы, доклад по результатам самостоятельной работы и т.д.);
- письменная (письменный опрос, выполнение, расчетно-проектировочной и расчетно-графической работ и т.д.);
- тестовая (устное, письменное, компьютерное тестирование).

Результаты текущего контроля успеваемости фиксируются в журнале занятий с соблюдением требований по его ведению.

Промежуточная аттестация – это элемент образовательного процесса, призванный определить соответствие уровня и качества знаний, умений и навыков обучающихся, установленным требованиям согласно рабочей программе дисциплины. Промежуточная аттестация осуществляется по результатам текущего контроля.

Конкретный вид промежуточной аттестации по дисциплине определяется рабочим учебным планом и рабочей программой дисциплины.

Зачет, как правило, предполагает проверку усвоения учебного материала практические и семинарских занятий, выполнения лабораторных, расчетно-проектировочных и расчетно-графических работ, курсовых проектов (работ), а также проверку результатов учебной, производственной или преддипломной практик. В отдельных случаях зачеты могут устанавливаться по лекционным курсам, преимущественно описательного характера или тесно связанным с производственной практикой, или имеющим курсовые проекты и работы.

Экзамен, как правило, предполагает проверку учебных достижений обучаемых по всей программе дисциплины и преследует цель оценить полученные теоретические знания, навыки самостоятельной работы, развитие творческого мышления, умения синтезировать полученные знания и их практического применения.

5. Материалы для оценки знаний, умений, навыков и (или) опыта деятельности

Полный комплект оценочных средств для оценки знаний, умений и навыков находится у ведущего преподавателя.