

**ГАОУ ВО «ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ НАРОДНОГО ХОЗЯЙСТВА»**

*Утверждены решением
Ученого совета ДГУНХ,
протокол № 9
от 21 марта 2025 г*

**КАФЕДРА «ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ИН-
ФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ»**

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

**ПО ДИСЦИПЛИНЕ
«ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И
ПРОГРАММИРОВАНИЕ»**

**НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.03 ПРИКЛАДНАЯ
ИНФОРМАТИКА,
ПРОФИЛЬ «ИНФОРМАЦИОННЫЕ СИСТЕМЫ
В ЭКОНОМИКЕ»**

Уровень высшего образования - бакалавриат

УДК 681.3.06

ББК 32.973.2-018 П784

Составители – Раджабов Карахан Якубович, кандидат экономических наук, доцент, доцент кафедры «Информационные технологии и информационная безопасность» ДГУНХ; Савина Елена Владимировна, кандидат физико-математических наук, доцент кафедры «Информационные технологии и информационная безопасность» ДГУНХ.

Внутренний рецензент – Кобзаренко Дмитрий Николаевич, доктор технических наук, профессор кафедры «Информационные технологии и информационная безопасность» ДГУНХ

Внешний рецензент – Абдурагимов Гусейн Эльдарханович, кандидат физико-математических наук, доцент кафедры "Математические методы в экономике" Дагестанского государственного университета.

Представитель работодателя - Сайидахмедов Сайидахмед Сергеевич, генеральный директор компании «Текама».

Оценочные материалы по дисциплине «Информационные технологии и программирование» разработаны в соответствии с требованиями федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.03 «Прикладная информатика», утвержденного приказом Министерства образования и науки Российской Федерации от 19 сентября 2017 г., №922, в соответствии с приказом Министерства образования и науки Российской Федерации от 06.04.2021 г. № 245 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры»

Оценочные материалы по дисциплине «Информационные технологии и программирование» размещены на официальном сайте www.dgunh.ru

Раджабов К.Я., Савина Е.В. Оценочные материалы по дисциплине «Информационные технологии и программирование» для направления подготовки 09.03.03 Прикладная информатика, профиль «Информационные системы в экономике». – Махачкала: ДГУНХ, 2025. - 45 с.

Рекомендованы к утверждению Учебно-методическим советом ДГУНХ 10 марта 2025 г.

Рекомендованы к утверждению руководителем основной профессиональной образовательной программы высшего образования – программы бакалавриата по направлению подготовки 09.03.03 Прикладная информатика, профиль «Информационные системы в экономике», к.э.н., доцентом Раджабовым К.Я.

Одобрены на заседании кафедры «Информационные технологии и информационная безопасность» 24 февраля 2025 г., протокол № 7.

СОДЕРЖАНИЕ

Назначение оценочных материалов.....	4
Раздел 1. Перечень компетенций с указанием видов оценочных средств в процессе освоения дисциплины.....	5
1.1. Перечень формируемых компетенций.....	5
1.2. Перечень компетенций с указанием видов оценочных средств.....	5
Раздел 2. Задания, необходимые для оценки планируемых результатов обучения по дисциплине.....	9
Раздел 3. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания.....	32
Раздел 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков, характеризующих этапы формирования компетенций.....	35
Лист актуализации оценочных материалов по дисциплине	45

Назначение оценочных материалов

Оценочные материалы для текущего контроля успеваемости (оценивания хода освоения дисциплин), для проведения промежуточной аттестации (оценивания промежуточных и окончательных результатов обучения по дисциплине) обучающихся по дисциплине «Информационные технологии и программирование» на соответствие их учебных достижений поэтапным требованиям образовательной программы высшего образования 09.03.03 Прикладная информатика, профиль «Информационные системы в экономике».

Оценочные материалы по дисциплине «Информационные технологии и программирование» включают в себя: перечень компетенций с указанием видов оценочных средств в процессе освоения дисциплины; описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания; типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения ОПОП; методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Оценочные материалы сформированы на основе ключевых принципов оценивания:

- валидности: объекты оценки должны соответствовать поставленным целям обучения;
- надежности: использование единообразных стандартов и критериев для оценивания достижений;
- объективности: разные обучающиеся должны иметь равные возможности для достижения успеха.

Основными параметрами и свойствами оценочных материалов являются:

- предметная направленность (соответствие предмету изучения конкретной дисциплины);
- содержание (состав и взаимосвязь структурных единиц, образующих содержание теоретической и практической составляющих дисциплины);
- объем (количественный состав оценочных материалов);
- качество оценочных материалов в целом, обеспечивающее получение объективных и достоверных результатов при проведении контроля с различными целями.

-

РАЗДЕЛ 1. Перечень компетенций с указанием видов оценочных средств в процессе освоения дисциплины

1.1 Перечень формируемых компетенций

код компетенции	формулировка компетенции
ОПК	ОБЩЕПРОФЕССИОНАЛЬНЫЕ КОМПЕТЕНЦИИ
ОПК-7	Способен разрабатывать алгоритмы и программы, пригодные для практического применения

1.2. Перечень компетенций с указанием видов оценочных средств

Формируемые компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, характеризующие этапы формирования компетенций	Уровни освоения компетенций	Критерии оценивания сформированности компетенций	Виды оценочных средств
ОПК-7: Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ИОПК-7.2. Разрабатывает алгоритмы и программы для практического применения	ЗНАТЬ: - процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов (информационные технологии); - логику построения и принципы функционирования современных языков программирования и языков работы с базами данных, среды	Пороговый уровень	Обучающийся слабо (частично) знает процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации, логику построения и принципы функционирования современных языков программирования и языков работы с базами данных, среды разработки информационных систем.	Блок А – задания репродуктивного уровня – тестовые задания
			Базовый уровень	Обучающийся с незначительными ошибками и отдельными пробелами знает процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации; - логику построения и принципы функционирования современных языков программирования и языков работы с базами дан-	

		<p>разработки информационных систем и технологий, принципы разработки алгоритмов и компьютерных программ;</p> <p>- современные языки программирования и языки работы с базами данных, среды разработки информационных систем и технологий.</p>		<p>ных, среды разработки информационных систем и технологий, принципы разработки алгоритмов и компьютерных программ;</p> <p>-современное языки программирования и языки работы с базами данных.</p>	
		<p>УМЕТЬ:</p> <p>-выбирать языки программирования и языки работы с базами данных, среды разработки информационных систем и технологий, исходя из имеющихся задач;</p> <p>-применять современные</p>	<p>Продвинутый уровень</p> <p>Пороговый уровень</p>	<p>Обучающийся с требуемой степенью полноты и точности знает - процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов (информационные технологии);</p> <p>-логику построения и принципы функционирования современных языков программирования и языков работы с базами данных, среды разработки информационных систем и технологий, принципы разработки алгоритмов и компьютерных программ;</p> <p>-современное языки программирования и языки работы с базами данных, среды разработки информационных систем и технологий.</p> <p>Обучающийся слабо (частично) умеет выбирать языки программирование и языки работы с базами данных, среды разработки информационных систем и технологий;</p> <p>-применять языки программирования, используя типовые алгоритмы, пригодные для практического применения, вести базы данных;</p> <p>-читать коды программных продуктов, написанных на освоенных языках</p>	<p>Блок В – задания реконструктивного уровня – лабораторные работы</p>

		языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения, вести базы данных и информационные хранилища, применять современные программные среды разработки информационных систем и технологий; -читать коды программных средств, написанных на освоенных языках программирования, и вносить требуемые изменения		программирования и вносить требуемые изменения	
			Базовый уровень	Обучающийся с незначительными затруднениями умеет -выбирать языки программирование и языки работы с базами данных, среды разработки информационных систем и технологий, исходя из имеющихся задач; -применять современные языки программирования и компьютерные программы, пригодные для практического применения, вести базы данных и информационные хранилища, применять современные программные среды разработки информационных систем и технологий; -читать коды программных продуктов, написанных на освоенных языках программирования, и вносить требуемые изменения	
			Продвинутый уровень	Обучающийся умеет выбирать языки программирование и языки работы с базами данных, среды разработки информационных систем и технологий, исходя из имеющихся задач; -применять современные языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения, вести базы данных и информационные хранилища, применять современные программные среды разработки информационных систем и технологий; -читать коды программных продуктов, написанных на освоенных языках	

				<p>программирования, и внести требуемые изменения;</p> <p>анализировать профессиональные задачи, разрабатывать подходящие ИТ-решения;</p>	
		<p>ВЛАДЕТЬ:</p> <p>- навыками разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического использования;</p> <p>- навыками отладки и тестирования прототипов программно—технических комплексов задач.</p>	<p>Пороговый уровень</p>	<p>Обучающийся слабо (частично) владеет навыками разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического использования;</p> <p>навыками отладки и тестирования прототипов программно — технических комплексов задач.</p>	<p>Блок С – задания практико-ориентированного уровня выполнение проекта;</p> <p>- комплексные задания для индивидуального и группового выполнения (на программирование)</p>
	<p>Базовый уровень</p>		<p>Обучающийся с небольшими затруднениями владеет навыками разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического использования;</p> <p>навыками отладки и тестирования прототипов программно — технических комплексов задач.</p>		
	<p>Продвинутый уровень</p>		<p>Обучающийся свободно владеет навыками разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического использования;</p> <p>навыками отладки и тестирования прототипов программно—технических комплексов задач.</p>		

РАЗДЕЛ 2. Задания, необходимые для оценки планируемых результатов обучения по дисциплине

Для проверки сформированности компетенции

ОПК-7: Способен разрабатывать алгоритмы и программы, пригодные для практического применения.

ИОПК-7.2. Разрабатывает алгоритмы и программы для практического применения.

Блок А. Задания репродуктивного уровня («знать»)

А.1 Фонд примерных тестовых заданий по дисциплине

Тесты типа А.

Алфавит, идентификаторы, операции

1. Зарезервированными словами в PascalABC.net являются

- 1) div 2) array 3) at 4) when
5) string 6) sum 7) unit 8) while

2. Зарезервированными словами в PascalABC.net являются

- 1) it 2) label 3) case 4) subject
5) end 6) foreach 7) some 8) go

3. Зарезервированными словами в PascalABC.net являются

- 1) set 2) kind 3) record 4) do
5) dim 6) loop 7) go 8) every

4. Зарезервированными словами в PascalABC.NET не являются

- 1) at 2) for 3) type 4) unit
5) some 6) while 7) sum 8) when

5. Зарезервированными словами в PascalABC.NET не являются

- 1) mod 2) at 3) until 4) when
5) function 6) loop 7) all 8) while

6. Установить соответствие

1	**	A	определение остатка от деления
2	/	B	уменьшение на ...
3	mod	C	возведение в степень
4	*	D	целочисленное деление
5	-=	E	умножение
6	div	F	деление

7. Установить соответствие

1	<>	A	логическое “и”
2	+=	B	условие “не равно”
3	or	C	целочисленное деление
4	mod	D	увеличение на ...
5	and	E	логическое “или”
6	div	F	определение остатка от деления

8. Выберите все верные утверждения. В PascalABC.net идентификатор
- 1) может начинаться с цифры
 - 2) может содержать латинские буквы
 - 3) может начинаться со знака подчеркивания
 - 4) может содержать пробелы
 - 5) не может иметь длину более 40 символов
9. Выберите все верные утверждения. В PascalABC.net идентификатор
- 1) может содержать цифры в любой позиции, кроме первой
 - 2) не может быть именем типа данных
 - 3) может содержать не более одного пробела
 - 4) может быть именем функции
 - 5) может содержать скобки
10. Выберите все верные утверждения
- 1) Операция not является бинарной
 - 2) Описание переменных может находиться в разделе операторов
 - 3) Имя константы не может начинаться с цифры
 - 4) В PascalABC.net символ ** обозначает возведение в степень
11. Выберите все верные утверждения
- 1) Зарезервированное слово нельзя использовать в качестве имени переменной
 - 2) Имя идентификатора не может начинаться со знака подчеркивания
 - 3) Порядок следования разделов описания переменных, констант, типов и меток в PascalABC.net строго определен
 - 4) Приоритет унарных операций выше, чем операций сравнения
12. Выберите все верные утверждения
- 1) Пару символов, обозначающих операцию «больше или равно» следует записывать через пробел
 - 2) Метки предназначены для использования в операторе безусловного перехода
 - 3) В PascalABC.net отсутствует операция возведения в степень
 - 4) В PascalABC.net допускается множественное присваивание значений переменным
13. Выберите все верные утверждения
- 1) Имя программы не может содержать пробелов
 - 2) В PascalABC.net отсутствует функция tg x
 - 3) В PascalABC.net константы можно описывать в разделе операторов
 - 4) Обращение к функции может быть операндом в выражении

Стандартные функции

1. К вещественным аргументам применимы функции

- 1) $\text{abs}(x)$ 2) $\text{pred}(x)$ 3) $\text{trunc}(x)$
4) $\text{exp}(x)$ 5) $\text{dec}(x)$ 6) $\text{int}(x)$

2. К вещественным аргументам неприменимы функции

- 1) $\tan(x)$ 2) $\text{succ}(x)$ 3) $\text{inc}(x)$
4) $\arcsin(x)$ 5) $\text{random}(x)$ 6) $\text{int}(x)$

3. Результат действия функции $\text{odd}(x)$ имеет тип

- 1) целый 2) символьный 3) логический 4) вещественный

4. Результат действия функции $\text{ord}(x)$ имеет тип

- 1) целый 2) символьный 3) логический 4) вещественный

5. Результат действия функции $\text{chr}(x)$ имеет тип

- 1) целый 2) символьный 3) логический 4) вещественный

6. Результат действия функции $\text{int}(x)$ имеет тип

- 1) целый 2) символьный 3) логический 4) вещественный

7. Установить соответствие

1	$\text{frac}(x)$	A	целая часть x
2	$\text{trunc}(x)$	B	округление до ближайшего целого
3	$\text{round}(x)$	C	дробная часть x
4	$\text{int}(x)$	D	отсечение дробной части

8. Установить соответствие

1	$\text{succ}(x)$	A	уменьшение x на 1
2	$\text{pred}(x)$	B	следующее значение x
3	$\text{inc}(x)$	C	увеличение x на 1
4	$\text{dec}(x)$	D	предыдущее значение x

9. Установить соответствие

1	$\text{abs}(x)$	A	экспонента x
2	$\tan(x)$	B	знак x
3	$\text{power}(x, y)$	C	модуль x
4	$\text{exp}(x)$	D	случайное целое из диапазона $[0; x)$
5	$\text{random}(x)$	E	тангенс x
6	$\text{sign}(x)$	F	возведение в степень

10. Установить соответствие между функцией и типом возвращаемого результата

1	$\text{odd}(x)$	A	целый
2	$\text{trunc}(x)$	B	вещественный
3	$\text{frac}(x)$	C	логический
4	$\text{chr}(x)$	D	символьный

5	int(x)	E	
6	sqrt(x)	F	
7	ord(x)	G	

Простые типы данных. Описание переменных

1. К целому типу данных не относятся

- 1) shortint 2) byte 3) single 4) word
5) char 6) double 7) longword 8) decimal

2. К вещественному типу данных относятся

- 1) byte 2) double 3) word
4) longint 5) decimal 6) single

3. Порядковыми типами являются

- 1) целый 2) символьный 3) вещественный
4) строковый 5) множественный

4. К целым типам относятся

- 1) double 2) uint64 3) word 4) integer
5) char 6) single 7) shortint 8) decimal

5. К вещественным типам относятся

- 1) double 2) decimal 3) word 4) integer
5) real 6) single 7) record 8) byte

6. Установить соответствие

1	символьный	A	set
2	массив	B	integer
3	целый	C	string
4	вещественный	D	array
5	множество	E	real
6	строка	F	char

7. Установить соответствие

1	логический	A	array
2	массив	B	set
3	символьный	C	record
4	запись	D	char
5	множество	E	string
6	строка	F	boolean

8. Установить соответствие

1	double	A	целый
2	byte	B	вещественный
3	longword		
4	decimal		
5	single		
6	smallint		
7	uint64		
8	integer		

9. Выберите правильные описания констант

- 1) `e = 2.72;` 2) `zero := 0;` 3) `at = 'at';`
4) `Language: 'Pascal';` 5) `no := false;` 6) `yes = true;`

10. Выберите неправильные описания констант

- 1) `al: const;` 2) `at = 'at';` 3) `no = false;`
4) `day = 'Monday';` 5) `Language = Pascal;` 6) `m = 1..6;`

Ввод-вывод, присваивание

1. Результат действия оператора `write('x=', x)`

- 1) ввод значения `x`
2) вывод символов `'x ='` и ввод значения `x`
3) вывод значения `x`
4) вывод символов `'x='` и значения `x`

2. Результат действия последовательности операторов `print('x=');` `read(x)`

- 1) ввод значения `x` 2) вывод символов `'x ='` и ввод значения `x`
3) вывод значения `x` 4) вывод символов `'x='` и значения `x`

3. Результат действия оператора `read(x)`

- 1) ввод значения `x`
2) вывод символов `'x ='` и ввод значения `x`
3) вывод значения `x`
4) вывод символов `'x='` и значения `x`

4. Результат действия оператора `println(x)`

- 1) ввод значения `x`
2) вывод символов `'x ='` и ввод значения `x`
3) вывод значения `x`
4) вывод символов `'x='` и значения `x`

5. Процедура `readln(x, y, ...)` предназначена для

- 1) вывода на экран строки символов `x y ...`
2) ввода значений переменных `x, y, ...` из файла
3) ввода с клавиатуры строки
4) ввода с клавиатуры значений переменных `x, y, ...`
5) вывода на экран значений переменных `x, y, ...`

6. Установить соответствие

1	read	A	ВВОД
2	print	B	ВЫВОД
3	readinteger		
4	readchar		
5	println		
6	writeln		
7	readreal2		
8	writeformat		

7. Установить соответствие

1	writeln('x =', x)	A	12 22
2	print(a, b)	B	x =15.42
3	println('x =', x)	C	x y
4	writeln('x =', x:5:2)	D	x =22
5	println('x', 'y')	E	xy
6	writeln('x', 'y')	F	x = 22

8. Установить соответствие

1	ReadInteger3	A	значение логического типа
2	Readreal2	B	значение вещественного типа
3	ReadReal	C	3 значения целого типа
4	ReadChar	D	3 строки
5	ReadBoolean	E	значение символьного типа
6	ReadString3	F	2 значения вещественного типа

9. Установить соответствие

1	Writeln(x, y)	A	ввод значений x и y
2	Print(x, y)	B	вывод значений x и y
3	Readln	C	перевод курсора в новую строку
4	Read(x, y)	D	вывод последовательности символов x y
5	Writeln	E	вывод последовательности символов xy
6	Println('x', 'y')	F	пауза в выполнении программы
7	Write('x', 'y')		
8	Println		

10. Функциями ввода данных в PascalABC.net являются
- 1) Println 2) ReadReal3 3) Read 4) WriteFormat
 - 5) ReadBoolean 6) Prtint 7) ReadChar

Операторы ветвления

1. Для организации алгоритма ветвления применяются операторы
 - 1) цикла с параметром 2) условный 3) выбора
 - 4) безусловного перехода 5) цикла с постусловием
2. Для организации алгоритма ветвления применяются операторы
 - 1) repeat 2) loop 3) case
 - 4) foreach 5) if 6) while
3. В результате выполнения фрагмента программы
a := -4; b := 1;
if a < 0 then b += a else b := 0;
переменная b принимает значение __
4. В результате выполнения фрагмента программы
a := 8; b := 0;
if 3*a < 25 then b -= a else b := a;
переменная b принимает значение __
5. В результате выполнения фрагмента программы
a := -1; b := 0; c := 3;
if 3*a = b + c then b := a*c else b := a + c;
переменная b принимает значение __

Операторы цикла

1. В операторе цикла с предусловием тело цикла
 - 1) обязательно выполняется более 1 раза
 - 2) выполняется по крайней мере 1 раз
 - 3) может не выполниться вообще
 - 4) выполняется строго определенное число раз
2. В операторе цикла с постусловием тело цикла
 - 1) обязательно выполняется более 1 раза
 - 2) выполняется по крайней мере 1 раз
 - 3) может не выполниться вообще
 - 4) выполняется строго определенное число раз
3. В операторе цикла с параметром for тело цикла
 - 1) обязательно выполняется более 1 раза
 - 2) выполняется по крайней мере 1 раз
 - 3) может не выполниться вообще
 - 4) выполняется строго определенное число раз

4. Установить правильную последовательность

- 1 **until**
- 2 *тело цикла*
- 3 **repeat**
- 4 *условие*

5. Установить правильную последовательность

- 1 *тело цикла*
- 2 *условие*
- 3 **while**
- 4 **do**

Структурированные типы данных

1. К структурированным типам относятся

- 1) вещественный
- 2) файл
- 3) множество
- 4) символьный
- 5) диапазон
- 6) целый

2. К структурированным типам относятся

- 1) символьный
- 2) строка
- 3) массив
- 4) диапазон
- 5) логический
- 6) запись

3. К структурированным типам не относятся

- 1) вещественный
- 2) множество
- 3) символьный
- 4) диапазон
- 5) массив
- 6) файл

4. Выберите все верные утверждения.

Тип данных определяет

- 1) формат их внутреннего представления в памяти компьютера
- 2) множество значений, которые может принимать данный объект
- 3) количество объектов в программе, которые могут принимать данные значения
- 4) количество символов в идентификаторе, присваиваемом объекту данного типа
- 5) множество допустимых операций, применяемых к данным

5. Выберите все верные утверждения.

- 1) логический тип данных относится к простым
- 2) строковый тип данных относится к простым
- 3) символьный тип данных относится к структурированным
- 4) указатель относится к структурированным типам данных
- 5) перечислимый тип данных относится к простым

Массивы

1. Выберите все правильные описания массивов

- 1) `var b: array [-2..8] of integer;`
- 2) `var m: array [20..1] of char;`
- 3) `var day: array [1..'n'] of char;`

- 4) var ch: array ['A'..'T'] of byte;
 5) var letter: array ['m'..'b'] of integer;
 6) var ord: array ['b'..'m'] of real;
2. Выберите все правильные описания массивов
- 1) var b: array ['a'..10] of integer;
 2) var map: array ['a'; 'd'] of boolean;
 3) var yes: array [10..20] of char;
 4) var chili: array ['A'..'N'] of real;
 5) var let: array [-8..-1] of boolean;
 6) var d: array ['d'...'p'] of real;
3. Выберите все правильные описания массивов
- 1) var dis: array ['a'..'n'] of real;
 2) var read: array ['t'..'a'] of integer;
 3) var may: array [10..7] of char;
 4) var base: array ['a'..'122'] of boolean;
 5) var d: array ['1'..'7'] of real;
 6) var abs: array [-10..0] of single;
4. Выберите все правильные описания массивов
- 1) var ch: array ['A'..'T'] of byte;
 2) var map: array ['a'; 'd'] of boolean;
 3) var n: array [-10..0] of single;
 4) var m: array [20..1] of char;
 5) var chili: array ['A'..'N'] of real;
 6) var s1: array ['a'..'122'] of boolean;
5. Фрагмент кода
- ```
var s := a[0];
for var i := 1 to 9 do if a[i] > s then s := a[i];
```
- вычисляет
- 1) максимальный элемент в массиве из 9 элементов  
 2) минимальный элемент в массиве из 10 элементов  
 3) максимальный элемент в массиве из 10 элементов  
 4) минимальный элемент в массиве из 9 элементов

### Двумерные массивы

1. Задан двумерный массив 5×5. В результате выполнения фрагмента программы
- ```
for var i := 1 to 5 do
  for var j := 1 to 5 do
    a[i, j] := i*j - 7;
```
- значение элемента a_{25} будет равно

2. Задан двумерный массив 5×6 . В результате выполнения фрагмента программы

```
for var i := 1 to 5 do
  for var j := 1 to 6 do
    a[i, j] := i + 2*j;
```

значение элемента a_{53} будет равно

3. Задан двумерный массив 4×6 . В результате выполнения фрагмента программы

```
for var i := 1 to 4 do
  for var j := 1 to 6 do
    a[i, j] := (i + j)*3;
```

значение элемента a_{51} будет равно

4. Фрагмент программы

```
for var i := 1 to 5 do
  for var j := 1 to 7 do
    read(a[i, j]);
```

определяет ввод массива

- | | |
|--------------------------------------|--------------------------------------|
| 1) размером 5×7 по столбцам | 2) размером 5×7 по строкам |
| 3) размером 7×5 по строкам | 4) размером 7×5 по столбцам |

5. Фрагмент программы

```
for var i := 1 to 7 do
  for var j := 1 to 5 do
    read(a[i, j]);
```

определяет ввод массива

- | | |
|--------------------------------------|--------------------------------------|
| 1) размером 5×7 по столбцам | 2) размером 5×7 по строкам |
| 3) размером 7×5 по строкам | 4) размером 7×5 по столбцам |

Строки, множества

1. Установить соответствие

1	concat(s1, s2)	A	вставка подстроки в строку
2	copy(s1, n, m)	B	удаление подстроки
3	delete(s1, n, m)	C	сцепление строк
4	insert(st1, st2, n)	D	отыскание подстроки в строке

2. Установить соответствие

1	+	A	включение элемента в множество
2	exclude	B	проверка принадлежности элемента множеству
3	in	C	исключение элемента из множества
4	-		
5	include		

3. **begin**

```
var s := ReadString;  
var t := ReadString;  
println(length(s) + length(t))
```

end.

Программа выводит на экран

- 1) количество символов в более длинной из двух строк
- 2) количество символов в более короткой из двух строк
- 3) число символов, на которое одна строка длиннее другой
- 4) суммарное количество символов в двух строках

4. **begin**

```
var s := ReadString;  
var t := ReadString;  
writeln(s + t);  
writeln(t + s)
```

end.

Программа выводит на экран

- 1) строку, полученную сцеплением строк s и t, из которой удалены повторяющиеся символы
- 2) строку, полученную в результате сцепления строк s и t, и ее длину
- 3) две строки – сначала s, потом t
- 4) две строки – результат сцепления s и t и результат сцепления t и s

5. **begin**

```
var s := ReadString;  
var t := ReadString;  
var m := length(s);  
var n := length(t);  
if m < n then writeln(m)  
else writeln(n);
```

end.

Программа выводит на экран

- 1) количество символов в более длинной из двух строк
- 2) количество символов в более короткой из двух строк
- 3) число символов, на которое одна строка длиннее другой
- 4) суммарное количество символов в двух строках

Файлы

1. Правильно описаны переменные файлового типа
 - 1) var dik: file
 - 2) var dik = file of char
 - 3) var dik = text
 - 4) var dik: file of byte
 - 5) var dik: file of string
 - 6) var dik = file of text
2. Правильно описаны переменные файлового типа
 - 1) var dik: file
 - 2) var dik = file of record
 - 3) var dik: text
 - 4) var dik: file of integer
 - 5) var dik: file of file
 - 6) var dik = text
3. Правильно описаны файловые типы данных
 - 1) type dik: file
 - 2) type dik = file of char
 - 3) type dik: text
 - 4) type dik = file
 - 5) type dik = text
 - 6) type dik = file of file
4. Правильно описаны файловые типы данных
 - 1) type dik = file
 - 2) type dik = file of string
 - 3) type dik: text
 - 4) type dik: file of file
 - 5) type dik = text
 - 6) type dik: file of char
5. Выберите допустимые описания типизированных файлов
 - 1) file of string
 - 2) file of char
 - 3) file of file
 - 4) file of record

Подпрограммы

1. Заголовок функции записан правильно
 - 1) function ter(var a, b = real, n = integer)
 - 2) function ter(a: real, b: real, n: integer): real
 - 3) function ter(var a: real; b: real; n: integer): longint
 - 4) function ter(a, b: real; n: integer): integer
2. Заголовок функции записан неправильно:
 - 1) function ter(var a, b = real, n = integer): char
 - 2) function ter(a: real, b: real, n: integer)
 - 3) function ter(var a: real; b: real; n: integer): byte
 - 4) function ter(a, b: real; n: integer): real
3. Заголовок процедуры записан правильно
 - 1) procedure(a, b: real; m: char): char
 - 2) procedure sept(var a: real; const c: string)
 - 3) procedure sept(a, c: real; n: char)
 - 4) procedure sept(var b: real, const c = 'day')
4. Заголовок процедуры записан неправильно
 - 1) procedure(a, b: real; m: char): char
 - 2) procedure sept(var a: real; const c: string)

- 3) procedure sept(a, c: real; n: char)
 - 4) procedure sept(var b: real, const c = 'day'): real
5. Выберите правильное утверждение

Результатом обращения

- 1) к функции может быть число, символ, строка
- 2) к функции всегда является число, символ или массив
- 3) к процедуре может быть только число или запись
- 4) к процедуре может быть какое-либо действие.

Блок В. Задания реконструктивного уровня («уметь»)

В1. Лабораторные (примерные) работы.

1. Записать выражения на языке Паскаль

- | | | |
|----------------------------------|--------------------------|---------------------------------------|
| 1) $\frac{2^x - x}{\sqrt{4x+1}}$ | 2) x^{115} | 3) $x^{22} + 22^x$ |
| 4) $\sqrt[12]{a^5 + 4a}$ | 5) $\sqrt[a]{x^a + a^x}$ | 6) $\frac{-b + \sqrt{b^2 - 4ac}}{2a}$ |

2. Записать оператор присваивания в виде обычной математической формулы.

- 1) $y := \sin(\text{abs}(\text{sqr}(\ln(x)))) / \text{sqrt}(x*x + x - 1)$
- 2) $y := \sin(\text{exp}(6.5*\ln(x))) / \cos(\text{exp}(6.5*\ln(x)))$
- 3) $y := x**\sin(x*x - 1)$
- 4) $y := (x + \sin(x))**x / \text{sqrt}(x*x*x - 1)$
- 5) $y := \cos(x**(x + 1)) - \sin(4*\ln(x))$

3. Пусть k и m – натуральные числа, k = 742 298. Определить значение переменной m после выполнения следующих операторов присваивания:

- 1) $m := k \text{ div } 100 \text{ mod } 10$
- 2) $m := k \text{ div } 100 \text{ div } 10$
- 3) $m := k \text{ mod } (100 \text{ div } 10)$
- 4) $m := k \text{ mod } 1000 \text{ mod } 10$
- 5) $m := k \text{ mod } 1000 \text{ div } 10$

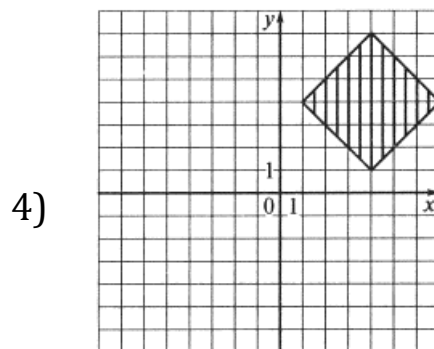
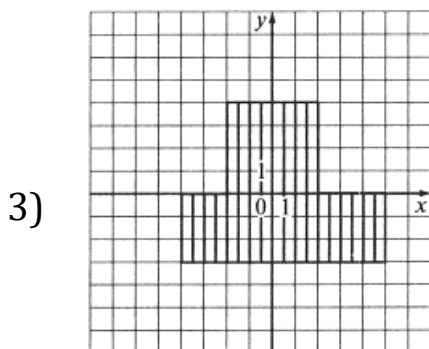
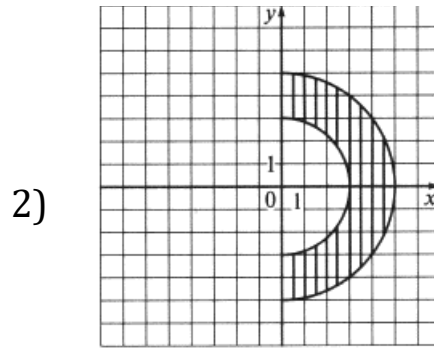
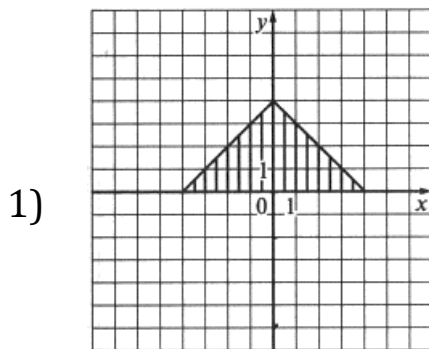
4) Записать оператор присваивания для следующих действий:

- 1) переменной x присвоить значение полуразности a и b;
- 2) удвоить значение переменной x;
- 3) поменять знак переменной x на противоположный;
- 4) уменьшить значение переменной целого типа x на 1;
- 5) заменить значение вещественной переменной t его абсолютной величиной.

5) Ввести величины двух катетов прямоугольного треугольника. Вывести на экран значения:

- 1) длины гипотенузы;

- 2) площади треугольника;
 3) меньшего из острых углов треугольника.
 б) Описать заштрихованные области с помощью логических выражений.



- 7) Написать код программы, которая выводит на экран значение true, если высказывание является истинным, и значение false в противном случае.

- сумма двух первых цифр данного четырехзначного числа равна сумме двух его последних цифр;
- сумма цифр заданного трехзначного числа является:
 - а) четным числом,
 - б) нечетным числом;
- сумма цифр данного трехзначного числа больше квадрата его первой цифры;

- 8) Фрагмент кода выполняет вычисление конечной суммы

$$\sum_{n=1}^7 (n^2 + 1) \cdot e^{-n} :$$

var s := 0.0;

for var n := 1 to 7 do

s += (n*n + 1)/exp(n);

Замените оператор цикла for оператором цикла

- а) с предусловием;
- б) с постусловием.

9) Для числовых последовательностей найти наименьший номер n , для которого выполняется неравенство $|a_n - a_{n-1}| < \varepsilon$ для заданного значения ε . Вывести на экран ε и n .

1) $a_n = 2 + \frac{1}{a_{n-1}}, a_1 = 2$

2) $a_n = \frac{1}{2^{n+1}}$

3) $a_n = \frac{1}{3} \operatorname{tg} a_{n-1}, a_1 = 0,5$

4) $a_n = \frac{1}{2} \cos a_{n-1}, a_1 = 0,5$

10) Определить количество членов последовательности натуральных чисел, кратных числу M и заключенных в интервале от L до N .

11) Дана прямоугольная матрица целых чисел A . Если она содержит отрицательные элементы, заменить их абсолютными величинами и вывести на экран новую матрицу. Если отрицательных элементов нет, вывести на экран матрицу такого же размера, как A , все элементы которой равны 1.

12) Создать список студентов из n строк (число n ввести с клавиатуры), содержащий следующую информацию: фамилия, курс, номер группы, оценка за экзамен. Вывести на экран фамилии студентов 3 группы 1 курса.

13) Множество содержит натуральные числа из некоторого диапазона. Сформировать из него два множества, первое из которых содержит все числа, кратные 3 или 7, а второе – все остальные числа.

14) Дан текст из строчных латинских букв, который заканчивается точкой. Вывести на экран

- все буквы, входящие в этот текст не менее двух раз;
- все буквы, входящие в этот текст по одному разу.

15) Заполнить файл f последовательного доступа целыми числами, полученными с помощью генератора случайных чисел. Записать в новый файл g компоненты файла f , являющиеся четными числами.

16) Разработать функцию, которая вычисляет значение наибольшего элемента одномерного массива действительных чисел с количеством элементов $n \leq 100$. Количество элементов в массиве должно задаваться при обращении к функции.

В2. Тематика (примерная) курсовых проектов

1. Функционал языка PascalABC.NET.
2. Аналогии языка PascalABC.NET и языка Python.
3. Целые типы в PascalABC.NET.
4. Типы вещественных чисел в PascalABC.NET.
5. Логика в программе, логический тип данных, логические выражения, операции отношения, логические операции.
6. Операторы - условный оператор, оператор выбора.
7. Цикл с заданным числом повторений (loop), цикл с параметром (for), цикл с предусловием (while), цикл с постусловием (repeat), изменение нормального хода выполнения цикла.
8. Подпрограммы в языке PascalABC.NET.
9. Последовательности в языке PascalABC.NET, генераторы.
- 10.Кортежи.
- 11.Множества.
- 12.Динамические массивы.
- 13.Символы и строки.
- 14.Записи.
- 15.Многомерные массивы.
- 16.Файлы.
- 17.Очередь (Queue).
- 18.Модули и библиотеки языка PascalABC.NET.
- 19.Обработка ошибок в программе на языке PascalABC.NET.
- 20.Объектно-ориентированное программирование, как технология программирования.

С1. Комплексные задания (примерные).

Var 1

Изучите программный код и выполните задания.

```
begin  
  var n := readinteger;  
  var k := 1;  
  while n div 10 <> 0 do  
    begin  
      n := n div 10;  
      k += 1  
    end;  
  println('k =', k)  
end.
```

1. В программном коде
 - 1) вводится действительное число n и определяется его знак
 - 2) вводится символьная переменная n и определяется ее код
 - 3) вводится натуральное число n и определяется сумма его цифр
 - 4) вводится целое число n и определяется количество цифр в нем
2. Перечислите переменные, которые использованы в коде, и назовите их типы
3. Если в коде используются выражения логического типа, назовите их.
4. Укажите операторы, которые использованы в коде
 - 1) оператор выбора
 - 2) оператор цикла с параметром
 - 3) оператор присваивания
 - 4) оператор цикла с постусловием
 - 5) условный оператор
 - 6) оператор цикла с предусловием
5. Перепишите программный код, заменив оператор `while ... do` оператором `repeat ... until`.

Var 02

Изучите программный код и выполните задания.

begin

```

var (a, b, c) := Read...;
var rs1 := (a + b > c) and (a + c > b) and (b + c > a);
var rs2 := (a = b) or (b = c) or (a = c);
if not rs1 then writeln('нет') else
  if rs2 then writeln('...')
end.
```

1. Фрагмент программного кода выполняет следующие действия
 - 1) определяет, существует ли треугольник со сторонами a , b , c
 - 2) вычисляет площадь треугольника со сторонами a , b , c
 - 3) вычисляет четвертую сторону описанного четырехугольника, три стороны которого равны a , b , c
 - 4) определяет, является ли треугольник прямоугольным
 - 5) определяет, является ли треугольник равнобедренным
 - 6) определяет, является ли треугольник тупоугольным или остроугольным
 - 7) вычисляет площадь равнобедренной трапеции с основаниями a и b и боковой стороной c

2. Выберите возможные продолжения, которые могут быть использованы в операторе присваивания во 2-й строке (вместо многоточия)

- 1) Real 2) Integer 3) Byte 4) Char
5) Boolean 6) Char3 7) Real3 8) String

3. Определите тип переменных rs1 и rs2.

4. В процедуре вывода в предпоследней строке вместо точек следует вписать текст

- 1) равносторонний 2) прямоугольный 3) тупоугольный
4) остроугольный 5) равнобедренный
6) треугольник не существует 7) неравносторонний
8) треугольник существует

5. Можно ли заменить условные операторы, использованные в коде, оператором выбора? Ответ обоснуйте.

Var 03

Изучите программный код и выполните задания.

begin

```
var (a, b) := Read...;
```

```
var h := Read...;
```

```
var x := a;
```

```
var y := 0.0;
```

```
repeat
```

```
  if x <= 1 then y := x*x - 1
```

```
  else y := ln(x);
```

```
  writeln('x = ', x:6:2, ' y = ', y:8:2);
```

```
  x += h
```

```
until x > b;
```

end.

1. Фрагмент программного кода реализует следующие действия:

- 1) вычисляет корень уравнения с заданной точностью
- 2) выводит на экран таблицу значений функции
- 3) вычисляет величины членов последовательности, не превосходящих заданного значения
- 4) выводит на экран значения последовательных приближений суммы ряда
- 5) выводит на экран значения членов последовательности, которые принадлежат заданному промежутку

2. Запишите формулу, которая используется в этом фрагменте, в обычной математической форме.
3. Оформите вычисление этой формулы в виде отдельной функции
4. Определите тип переменных h, a, b, x, y
5. Чем можно заменить оператор присваивания $x += h$? Укажите все возможности.
6. Замените оператор repeat ... until оператором while ... do

Var.04

Изучите программный код и выполните задания.

begin

```

var x := Read...;
var k: ...;
if x <= 1 then k := 0;
if (x > 1) and (x <= 5) then k := 1;
if x > 5 then k := 2;
var y: ...;
case k of
  1: y := sin(sqr(x));
  2: y := exp(2/x);
  3: y := sqrt(x*x - ln(x))
end;
Writeln ('x = ', x:5:2, ' y = ', y:6:2);

```

end.

1. Фрагмент программы выполняет следующие действия:
 - 1) выводит на экран значения x, не превосходящие 1
 - 2) вычисляет значение числовой функции и выводит его на экран
 - 3) определяет, принадлежит ли значение x промежутку (1; 5]
 - 4) вычисляет значение логической функции y и выводит его на экран
 - 5) вычисляет значения числовых функций x, y и k
2. Замените 3 условных оператора (строки 4-6) отдельной функцией или объясните, почему этого сделать нельзя.
3. Как следует изменить код, чтобы исключить переменную k?
4. Запишите формулу, которая использована в приведенном коде, в обычной математической форме.
5. Определите тип переменных k, x, y.;

Var 05

Изучите программный код и выполните задания.

```
const n = 20;
begin
  var a: array[1..n, 1..n] of integer;
  var b: ...;
  for var i := 1 to n do
    for var j := 1 to n do
      a[i, j] := random(300);
  for var i := 1 to n do
    begin
      var R := a[i, 1];
      for var j := 2 to n do
        if a[i, j] > R then R := a[i, j];
      b[i] := R
    end;
  for var i := 1 to n do print(b[i]);
end.
```

1. Код выполняет действия

- 1) заполняет одномерный массив случайными числами
- 2) 1-й элемент каждой строки записывает в одномерный массив и выводит этот массив на экран
- 3) заполняет случайными числами квадратную матрицу
- 4) определяет наименьший элемент в каждой строке матрицы
- 5) определяет наибольший элемент в каждой строке матрицы, записывает в новый массив и выводит его на экран
- 6) определяет наибольший элемент в каждом столбце матрицы, записывает его номер в новый массив и выводит его на экран
- 7) заносит в новый массив среднее арифметическое элементов каждой строки матрицы

2. Оформите условие $a[i, j] > R$ в виде функции и покажите, как изменится строка кода, отмеченная звездочкой, если использовать эту функцию.

3. Определите тип переменной b .

4. Выберите все верные утверждения

- 1) переменная a может иметь тип `array[1..n, 1..n] of char`
- 2) переменная a может иметь тип `array[1..n, 1..n] of real`
- 3) переменные a и b имеют один и тот же тип данных
- 4) матрица a заполнена случайными числами
- 5) условный оператор в строке, помеченной звездочкой, нельзя заменить оператором выбора
- 6) массив b содержит n элементов
- 7) массивы a и b имеют один и тот же базовый тип данных

5. Замените оператор цикла с параметром в предпоследней строке кода оператором цикла `foreach`.

Блок Д. Задания для использования в рамках промежуточной аттестации

Д1. Перечень (примерных) экзаменационных вопросов

1. Информация. Информатика. Информационные технологии. Информация и данные. Формы представления информации.
2. Системы счисления. Перевод чисел из одной системы счисления в другие.
3. Этапы развития вычислительной техники. Определение ЭВМ. Классификация ЭВМ. Обобщенная структурная схема ЭВМ. Устройства ввода ЭВМ. Назначение, типы.
4. Устройства вывода ЭВМ. Назначение, типы. Основная память ЭВМ. Назначение и состав. Внешние запоминающие устройства ЭВМ, Назначение и типы.
5. Центральные устройства ЭВМ. Состав и принцип работы. Обработка машинных команд центральными устройствами ЭВМ.
6. Взаимодействие центральных и внешних устройств ЭВМ. Типы интерфейса.
7. Шина. Характеристики и типы.
8. Обобщенная структурная схема персонального компьютера.
9. Программное обеспечение ЭВМ. Типы и состав.
10. Операционные системы. Основные функции и виды.
11. Типы диалога пользователь с компьютером.
12. Разработка прикладной программы под управлением ЭВМ.
13. Системы программирования. Назначение и состав.
14. Основы структурного программирования.
15. Базовые управляющие конструкции.
16. Алгоритм и схема алгоритма. Тестирование и отладка программ. Виды ошибок в программах.
17. Методы получения дополнительной информации о процессе выполнения программ.
18. Назначение и типы вычислительных комплексов.
19. Назначение и типы компьютерных сетей. Состав и основные характеристики компьютерных сетей. Виды топологии компьютерных сетей.
20. Сеть Интернет. Структура. Управление. Протоколы.
21. Адреса компьютера в сети Интернет. Унифицированный указатель ресурсов. Основные службы сети Интернет.

22. Типы данных. Объявление типа переменной. Оператор присваивания. Арифметические операции. Арифметические выражения. Математические функции. Примеры.
23. Организация ввода и вывода данных. Примеры.
24. Логические константы, переменные, операции, выражения. Примеры.
25. Программирование ветвлений. Условный оператор. Примеры.
26. Программирование повторений. Цикл со счетчиком. Примеры.
27. Программирование повторений. Циклы с условием. Примеры.
28. Организация программ со структурой вложенных циклов на примере
29. использования оператора цикла с постусловием.
30. Организация программ со структурой вложенных циклов на примере использования оператора цикла с предусловием.
31. Организация программ со структурой вложенных циклов на примере использования оператора цикла со счетчиком.
32. Приемы определения количества, формирования суммы, произведения, факториала.
33. Приемы нахождения наименьшего и наибольшего значений в последовательности, массивы.
34. Одномерные массивы. Ввод и вывод массивов. Примеры.
35. Динамические массивы. Объявление массива, изменение его размера. Примеры использования.
36. Формирование нового массива из элементов исходного массива.
37. Сортировка одномерных массивов. Пример.
38. Двумерные массивы. Объявление массива. Ввод матриц. Примеры.
39. Двумерные массивы. Обработка квадратных матриц. Примеры.
40. Строки. Основные функции обработки строк. Примеры.
41. Классификация подпрограмм. Процедуры. Передача параметров по значению и по ссылке. Глобальные и локальные переменные. Примеры.
42. Классификация подпрограмм. Функции. Определение возвращаемого значения функции. Примеры.

Дисциплина «**ИТ и программирование**» предполагает выполнение курсового проекта для систематизации, закрепления и углубления знаний, умений и навыков обучающихся в предметной области дисциплины.

Аттестация по курсовому проекту производится в виде ее защиты в рамках промежуточного контроля обучающихся по данной дисциплине во 2 семестре 1 курса в период теоретического обучения.

РАЗДЕЛ 3. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Балльно-рейтинговая система является базовой системой оценивания сформированности компетенций обучающихся очной формы обучения.

Итоговая оценка сформированности компетенции обучающихся в рамках балльно-рейтинговой системы осуществляется в ходе текущего контроля успеваемости, промежуточной аттестации и определяется как сумма баллов, полученных обучающимися в результате прохождения всех форм контроля.

Оценка сформированности компетенции по дисциплине складывается из двух составляющих:

✓ первая составляющая – оценка преподавателем сформированности компетенции в течение семестра в ходе текущего контроля успеваемости (максимум 100 баллов). Структура первой составляющей определяется технологической картой дисциплины, которая в начале семестра доводится до сведения обучающихся;

✓ вторая составляющая – оценка сформированности компетенции обучающихся на экзамене (максимум – 30 баллов)

Для студентов заочной и очно-заочной форм обучения применяется 4-балльная и бинарная шкалы оценивания результатов текущего контроля успеваемости и промежуточной аттестации обучающихся.

Уровни освоения компетенций	продвинутый уровень	базовый уровень	пороговый уровень	допороговый уровень
100 – балльная шкала	85 и \geq	70 – 84	51 – 69	0 – 50
4 – балльная шкала	«отлично»	«хорошо»	«удовлетворительно»	«неудовлетворительно»

Шкала оценок при текущем контроле успеваемости по различным показателям

<i>Показатели оценивания сформированности компетенций</i>	<i>Баллы</i>	<i>Оценка</i>
Выполнение лабораторных работ	0-20	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Проведение опроса	0-10	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Тестирование	0-30	«неудовлетворительно» «удовлетворительно» «хорошо»

		«отлично»
Выполнение и защита проекта (на программирование)	0-5	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»
Выполнение и публичная защита презентации	0-5	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»

Соответствие критериев оценивания уровню освоения компетенций по текущему контролю успеваемости

<i>Баллы</i>	<i>Оценка</i>	<i>Уровень освоения компетенций</i>	<i>Критерии оценивания</i>
0-50	«неудовлетворительно»	Допороговый уровень	Обучающийся не приобрел знания, умения и не владеет компетенциями в объеме, закрепленном рабочей программой дисциплины
51-69	«удовлетворительно»	Пороговый уровень	Не менее 50% заданий, подлежащих текущему контролю успеваемости, выполнены без существенных ошибок
70-84	«хорошо»	Базовый уровень	Обучающимся выполнено не менее 75% заданий, подлежащих текущему контролю успеваемости, или при выполнении всех заданий допущены незначительные ошибки; обучающийся показал владение навыками систематизации материала и применения его при решении практических заданий; задания выполнены без ошибок
85-100	«отлично»	Продвинутый уровень	100% заданий, подлежащих текущему контролю успеваемости, выполнены самостоятельно и в требуемом объеме; обучающийся проявляет умение обобщать, систематизировать материал и применять его при решении практических заданий; задания выполнены с подробными пояснениями и аргументированными выводами

Соответствие критериев оценивания уровню освоения компетенций по текущему контролю успеваемости

<i>Баллы</i>	<i>Оценка</i>	<i>Уровень освоения компетенций</i>	<i>Критерии оценивания</i>
0-50	«неудовлетворительно»	Допороговый уровень	Обучающийся не приобрел знания, умения и не владеет компетенциями в объеме, закрепленном рабочей программой дисциплины
51-69	«удовлетворительно»	Пороговый уровень	Не менее 50% заданий, подлежащих текущему контролю успеваемости, выполнены без существенных ошибок
70-84	«хорошо»	Базовый уровень	Обучающимся выполнено не менее 75% заданий, подлежащих текущему контролю успеваемости, или при выполнении всех заданий допущены незначительные ошибки; обучающийся показал владение навыками систематизации материала и применения его при решении практических заданий; задания выполнены без ошибок

Шкала оценок по промежуточной аттестации

<i>Наименование формы промежуточной аттестации</i>	<i>Баллы</i>	<i>Оценка</i>
Экзамен	0-30	«неудовлетворительно» «удовлетворительно» «хорошо» «отлично»

Соответствие критериев оценивания уровню освоения компетенций по промежуточной аттестации обучающихся

<i>Баллы</i>	<i>Оценка</i>	<i>Уровень освоения компетенций</i>	<i>Критерии оценивания</i>
0-9	«неудовлетворительно»	Допороговый уровень	Обучающийся не приобрел знания, умения и не владеет компетенциями в объеме, закрепленном

			рабочей программой дисциплины; обучающийся не смог ответить на вопросы
10-16	«удовлетворительно»	Пороговый уровень	Обучающийся дал неполные ответы на вопросы, с недостаточной аргументацией, практические задания выполнены не полностью, компетенции, осваиваемые в процессе изучения дисциплины сформированы не в полном объеме.
17-23	«хорошо»	Базовый уровень	Обучающийся в целом приобрел знания и умения в рамках осваиваемых в процессе обучения по дисциплине компетенций; обучающийся ответил на все вопросы, точно дал определения и понятия, но затрудняется подтвердить теоретические положения практическими примерами; обучающийся показал хорошие знания по предмету, владение навыками систематизации материала и полностью выполнил практические задания
25-30	«отлично»	Продвинутый уровень	Обучающийся приобрел знания, умения и навыки в полном объеме, закрепленном рабочей программой дисциплины; терминологический аппарат использован правильно; ответы полные, обстоятельные, аргументированные, подтверждены конкретными примерами; обучающийся проявляет умение обобщать, систематизировать материал и выполняет практические задания с подробными пояснениями и аргументированными выводами

РАЗДЕЛ 4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков, характеризующие этапы формирования компетенций

Тестирование проводится на семинарских занятиях. Самостоятельное выполнение обучающимся учебной группы в течение 30 минут индивидуального тестового задания.

Цель блока - формирование инструментальной компетенции использовать знания базового аппарата дисциплины для решения конкретных задач, самостоятельного приобретения знаний данной дисциплины в условиях повышения личностной мотивации выполнения работы.

Образовательными задачами блока являются:

- глубокое изучение лекционного материала, изучение методов работы с учебной литературой, получение персональных консультаций у преподавателя;
- решение спектра прикладных задач, в том числе профессиональных;
- работа с организационно - управленческими документами

На тестирование отводится 30 минут.

Методика оценивания выполнения тестов

Баллы	Оценка	Показатели	Критерии
25-30	«отлично»	1. Полнота выполнения тестовых заданий;	Выполнено 90-100 % заданий предложенного теста, в заданиях открытого типа дан полный, развернутый ответ на поставленный вопрос
17-24	«хорошо»	2. Своевременность выполнения;	
7-16	«удовлетворительно»	3. Правильность ответов на вопросы;	
0-15	«неудовлетворительно»	4. Самостоятельность тестирования;	
		5. и т.д.	
			Выполнено 80-89 % заданий предложенного теста, в заданиях открытого типа дан полный, развернутый ответ на поставленный вопрос; однако были допущены неточности в определении понятий, терминов и др.
			Выполнено 51-79 % заданий предложенного теста, в заданиях открытого типа дан неполный ответ на поставленный вопрос, в ответе не присутствуют доказательные примеры, текст со стилистическими и орфографическими ошибками.
			Выполнено 0-50% заданий предложенного теста, на поставленные вопросы ответ отсутствует или неполный, допущены существенные ошибки в теоретическом материале (терминах, понятиях).

Устный опрос позволяет оценить знания и кругозор студента, умение логически построить ответ, владение монологической речью и иные коммуникативные навыки. Проводятся преподавателем с обучающимся на темы, связанные с изучаемой дисциплиной, рассчитана на выяснение объема знаний, обучающегося по определенному разделу, теме, проблеме и т.п.

Методика оценивания ответов на устные вопросы

Баллы	Оценка	Показатели	Критерии
8-10	«отлично»	1. Полнота данных ответов; 2. Аргументированность данных ответов; 3. Правильность ответов на вопросы т.д.	Полно и аргументировано даны ответы по содержанию задания. Обнаружено понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные. Изложение материала последовательно и правильно.
6-7	«хорошо»		Студент дает ответ, удовлетворяющий тем же требованиям, что и для оценки «отлично», но допускает 1-2 ошибки, которые сам же исправляет.
3-5	«удовлетворительно»		Студент обнаруживает знание и понимание основных положений данного задания, но: 1) излагает материал неполно и допускает неточности в определении понятий или формулировке правил; 2) не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; 3) излагает материал непоследовательно и допускает ошибки.
0-2	«неудовлетворительно»		Студент обнаруживает незнание ответа на соответствующее задание, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал; отмечаются такие недостатки в подготовке студента, которые являются серьезным препятствием к успешному овладению последующим материалом.

Лабораторная работа — это форма организации учебного процесса, когда обучающиеся по заданию и под руководством преподавателя самостоятельно разрабатывают программные приложения, осуществляют настройку программной среды, проводят отладку программ, реализуют как типовые, так и собственные алгоритмы решения при выполнении предложенных заданий. Лабораторная работа как вид учебного занятия должна проводиться в специально оборудованных учебных лабораториях или компьютерных классах. Продолжительность - не менее двух академических часов. Необходимыми структурными элементами лабораторной работы, помимо самостоятельной деятельности студентов, являются инструктаж, проводимый преподавателем, а также организация обсуждения итогов выполнения лабораторной работы.

Содержание лабораторного занятия определяется перечнем формируемых компетенций по конкретной учебной дисциплине, а также характеристикой профессиональной деятельности выпускников, требованиями к результатам освоения основной профессиональной образовательной программы.

Защита лабораторной работы позволяет оценить умение и владение обучающегося излагать суть поставленной задачи, применять стандартные методы решения задачи с использованием имеющейся лабораторной базы, проводить анализ результата работы.

Методика оценивания выполнения лабораторных заданий

<i>Баллы</i>	<i>Оценка</i>	<i>Показатели</i>	<i>Критерии</i>
16-20	«отлично»	1. Полнота выполнения лабораторной работы; 2. Своевременность выполнения лабораторной работы;	Выполнены все требования к лабораторной работе; разработана, отлажена и протестирована программа; даны правильные ответы на дополнительные вопросы.
11-15	«хорошо»	3. Правильность выполнения лабораторной работы.	Выполнены основные требования к лабораторной работе, имеются недочеты в разработке и тестировании программы; на дополнительные вопросы при защите даны неполные ответы.
3-10	«удовлетворительно»		Требования к лабораторной работе выполнены не полностью, программа разработана, но в ней имеются существенные недостатки; допущены фактические ошибки при ответе на дополнительные вопросы.
0-2	«неудовлетворительно»		Лабораторная работа не выполнена; обнаруживается существенное непонимание в ее выполнении.

Индивидуальный (групповой) проект представляет собой особую форму организации деятельности обучающихся (учебное исследование или учебный проект).

Проект выполняется обучающимся самостоятельно под руководством преподавателя по выбранной теме в рамках дисциплины.

Проект выполняется обучающимся персонально или в составе группы в течение 1 семестра в рамках семинарских занятий, специально отведенной учебным планом, и должен быть представлен в виде завершеного программного приложения или разработанного проекта.

Процедура работы над проектом разбивается на 6 этапов:

- подготовительный (определение руководителей проектов, поиск проблемного поля, выбор темы и её конкретизация, формирование проектной группы)
- поисковый (уточнение тематического поля и темы проекта, её конкретизация, определение и анализ проблемы, постановка цели проекта)
- аналитический (анализ имеющейся информации, поиск информационных лакун, сбор и изучение информации, поиск оптимального способа достижения цели проекта, построение алгоритма деятельности, составление плана реализации проекта: пошаговое планирование работ, анализ ресурсов)
- практический (выполнение запланированных технологических операций, текущий контроль качества составления проекта, внесение (при необходимости) изменений в разработку проекта)
- презентационный (подготовка презентационных материалов, презентация проекта, изучение возможностей использования результатов проекта)
- контрольный (анализ результатов выполнения проекта - оценка качества выполнения проекта)

Методика оценивания выполнения индивидуальных (групповых) проектов

Баллы	Оценка	Показатели	Критерии
5	«отлично»	1. Полнота выполнения проекта; 2. Своевременность выполнения проекта;	Выполнены все требования к выполнению проекта; разработана, отлажена и протестирована программа; даны правильные ответы на дополнительные вопросы.
3-4	«хорошо»	3. Правильность выполнения проекта.	Выполнены основные требования к проекту, имеются недочеты в разработке и тестировании программы; на дополнительные вопросы при защите даны неполные ответы.
1-2	«удов-летворительно»		Требования к проекту выполнены

	влетво- ри- тельно»		не полностью, программа разрабо- тана, но в ней имеются существен- ные недостатки; допущены факти- ческие ошибки при ответе на до- полнительные вопросы.
0	«неудо- влетво- ри- тельно»		Проект не выполнен; обнаружива- ется существенное непонимание в том, как его выполнять.

Экзамен нацелен на комплексную проверку освоения дисциплины. Экзамен проводится в письменной форме по билетам, в которых содержатся вопросы (задания) по всем темам курса. Обучающемуся даётся время на подготовку. Оценивается владение материалом, его системное освоение, способность применять нужные знания, навыки и умения при анализе проблемных ситуаций и решении практических заданий.

В экзаменационный билет включено два теоретических вопроса и практическое задание на программирование, соответствующие содержанию формируемых компетенций. Экзамен проводится в устной форме, практическое задание выполняется с использованием персонального компьютера. На ответ и решение задачи студенту отводится 40 минут.

Методика оценивания ответа на экзамене

<i>Баллы</i>	<i>Оценка</i>	Показатели	Критерии
25-30	«отлично»	1. Полнота изложения теоретического материала; 2. Полнота и правильность решения практического задания; 3. Правильность и/или аргументированность изложения (последовательность действий);	Дан полный, в логической последовательности развернутый ответ на поставленный вопрос, где он продемонстрировал знания предмета в полном объеме учебной программы, достаточно глубоко осмысливает дисциплину, самостоятельно, и исчерпывающе отвечает на дополнительные вопросы, приводит собственные примеры по проблематике поставленного вопроса, решил предложенные практические задания без ошибок.

17-24	«хорошо»	4. Самостоятельность ответа; 5. Культура речи; 6. и т.д.	Дан развернутый ответ на поставленный вопрос, где студент демонстрирует знания, приобретенные на лекционных и семинарских занятиях, а также полученные посредством изучения обязательных учебных материалов по курсу, дает аргументированные ответы, приводит примеры, в ответе присутствует свободное владение монологической речью, логичность и последовательность ответа. Однако допускается неточность в ответе. Решил предложенные практические задания с небольшими неточностями.
10-16	«удовлетворительно»		Дан ответ, свидетельствующий в основном о знании процессов изучаемой дисциплины, отличающийся недостаточной глубиной и полнотой раскрытия темы, знанием основных вопросов теории, слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры, недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа и решении практических заданий.

Дисциплина «Информационные технологии и программирование» предполагает выполнение курсового проекта для систематизации, закрепления и углубления знаний, умений и навыков, обучающихся в предметной области дисциплины.

Аттестация по курсовому проекту/работе производится в виде ее защиты в рамках промежуточного контроля обучающихся по данной дисциплине во 2 семестре в период теоретического обучения. Примерная тематика **курсовых проектов** приведена в разделе **В2/**

Критерии оценки курсового проекта

При оценке курсовых проектов по дисциплине «Информационные технологии и программирование» оценка формируется на основе следующих показателей:

1. Методологические характеристики;
2. Результат работы;
3. Оформление работы.

1. Оценка методологических характеристик курсового проекта

Требования	Оценка
<p><i>Во введении:</i> <u>В основном</u> обоснована практическая актуальность темы для данной организации (ситуации) Указан <u>адекватный специальности</u> объект исследования Предмет курсовой работы/проекта <u>адекватен теме и цели</u>, указывает на аспект или часть объекта Цель курсовой работы/проекта <u>фиксирует</u> ожидаемые результаты работы, <u>адекватна</u> теме и предмету Последовательность поставленных задач <u>в целом</u> позволяет достичь цели; структура работы (оглавление) <u>в целом</u> адекватна поставленным задачам <i>В заключении:</i> В выводах представлены <u>основные результаты</u> работы</p>	Удовлетворительно
<p><i>Во введении:</i> Обоснована <u>практическая</u> актуальность темы для данной организации (ситуации), на основе анализа степени разработанности вопроса в литературе <u>в основном</u> обоснована теоретическая актуальность темы Указан адекватный специальности объект курсовой работы/проекта Предмет курсовой работы/проекта <u>адекватен теме и цели</u>, указывает на аспект или часть объекта Цель курсовой работы/проекта <u>фиксирует</u> ожидаемые результаты работы, <u>адекватна</u> теме и предмету Последовательность поставленных задач <u>в основном</u> позволяет достичь цели; структура работы (оглавление) <u>в основном</u> адекватна поставленным задачам <i>В заключении:</i> В выводах представлены <u>все результаты</u> работы</p>	Хорошо
<p><i>Во введении:</i> Обоснована <u>практическая</u> актуальность темы для данной организации (ситуации), на основе анализа степени разработанности вопроса в литературе аргументированно обоснована <u>теоретическая</u> актуальность темы</p>	Отлично

<p>Указан <u>адекватный направлению</u> объект курсовой работы/проекта</p> <p>Предмет курсовой работы/проекта <u>адекватен теме и цели</u>, указывает на аспект или часть объекта</p> <p>Цель курсовой работы/проекта <u>фиксирует</u> ожидаемые результаты работы, <u>адекватна</u> теме и предмету</p> <p>Последовательность поставленных задач <u>позволяет</u> достичь цели рациональным способом; структура работы (оглавление) <u>полностью</u> адекватна поставленным задачам</p> <p><i>В заключении:</i></p> <p>В выводах представлены <u>все результаты</u> работы в <u>наиболее адекватной</u> форме</p>	
--	--

2. Оценка *результатов*, полученных автором курсового проекта

Характеристики содержания и результатов работы	Оценка
<p>Полученные результаты <u>в значительной степени</u> соответствуют поставленной цели (цель работы достигнута в значительной степени)</p> <p>В процессе анализа литературы отобраны <u>в целом адекватные</u> источники, продемонстрировано <u>понимание</u> решаемой проблемы</p> <p>В конкретной (практической) ситуации выбрана <u>адекватная</u> цели управленческая технология (подход, инструмент, метод, процедура)</p> <p>Технология (подход, инструмент, метод, процедура) <u>описана</u></p> <p>Технология (подход, инструмент, метод, процедура) осуществлена (реализована) <u>в значительной степени</u></p>	Удовлетворительно
<p>Полученные результаты <u>преимущественно</u> соответствуют поставленной цели (цель работы преимущественно достигнута)</p> <p>В процессе анализа литературы отобраны <u>адекватные</u> источники, сделаны <u>адекватные</u> выводы</p> <p>Выбрана <u>адекватная</u> цели управленческая технология (подход, инструмент, метод, процедура), осуществлено <u>обоснование</u> выбора</p> <p>Технология (подход, инструмент, метод, процедура) <u>описана полностью</u></p> <p>Технология (подход, инструмент, метод, процедура) <u>преимущественно</u> осуществлена (реализована)</p>	Хорошо

<p>Полученные результаты <u>полностью</u> соответствуют поставленной цели (цель работы достигнута полностью)</p> <p>Проведен <u>детальный анализ</u> адекватных источников, выводы <u>самостоятельны и аргументированы</u></p> <p>Отобрана <u>наиболее адекватная</u> цели управленческая технология (подход, инструмент, метод, процедура), обоснование выбора <u>аргументировано</u></p> <p>Представлено <u>детальное описание</u> технологии (подхода, модели, инструмента, метода, процедуры)</p> <p>Технология (подход, инструмент, метод, процедура) осуществлена (реализована) <u>полностью</u></p>	Отлично
--	---------

3. Оценка оформления курсового проекта

Требования к оформлению выпускной квалификационной работы	Оценка
<ol style="list-style-type: none"> 1. Работа выполнена на стандартных листах бумаги формата А4, шрифт Times New Roman, кегль 14, интервал полуторный. Поля: сверху - 2 см, снизу - 2,5 см, справа - 1 см, слева 3 см 2. Страницы пронумерованы арабскими цифрами, номер по центру сверху страницы, титульный лист не пронумерован. Заголовки расположены по центру строк. В заголовках отсутствуют переносы. В конце заголовков отсутствуют знаки препинания. Отсутствуют заголовки в конце страниц. 3. Знаки препинания проставлены непосредственно после последней буквы слова и отделены от следующего слова пробелом. Абзацы напечатаны с красной строки, при этом от левого поля имеется отступ 1,25 см. 4. Все таблицы и рисунки имеют нумерацию и названия (над таблицей справа, под рисунком слева) 5. На все таблицы и рисунки даны ссылки в тексте 6. При использовании заимствованного материала применены ссылки причем однотипные (либо постраничные, либо концевые) 7. Список литературы составлен в алфавитном порядке 8. Описание каждого источника в списке литературы содержит фамилию (фамилии) автора (авторов), заглавие, место издания, год издания, либо дату, если издание периодическое, адрес web-страницы, если используются ресурсы Интернет (печатается в начале описания источника) 	

Оценка "отлично" ставится за оформление курсового проекта, полностью отвечающее представленным требованиям.

Оценка "хорошо" ставится за оформление курсового проекта, в целом отвечающее представленным требованиям, но при наличии отдельных отступлений не более чем по двум требованиям.

Оценка "удовлетворительно" ставится за оформление курсового проекта, в целом отвечающее представленным требованиям, но при наличии отдельных отступлений не более чем по трем требованиям.

Не может быть поставлена положительная оценка за оформление курсового проекта, если полностью не выполнены требования 1, 2, 7, 8, 10.

Итоговая оценка курсового проекта:

1. Оценка <i>методологических характеристик</i> курсовой работы/проекта	5	4-5	4	4-5	Хотя бы одна 3
2. Оценка <i>результатов</i> , полученных автором курсовой работы/проекта	5	4-5	4-5	4	
3. Оценка <i>оформления</i> курсовой работы/проекта	5	4	4-5	4-5	
Итоговая оценка	Отлично	Хорошо		Удовлетворительно	

Лист актуализации оценочных материалов по дисциплине
«Информационные технологии и программирование»

Оценочные материалы пересмотрены,
обсуждены и одобрены на заседании кафедры

Протокол от «_____» _____ 20__ г. № _____

Зав. кафедрой _____

Оценочные материалы пересмотрены,
обсуждены и одобрены на заседании кафедры

Протокол от «_____» _____ 20__ г. № _____

Зав. кафедрой _____

Оценочные материалы пересмотрены,
обсуждены и одобрены на заседании кафедры

Протокол от «_____» _____ 20__ г. № _____

Зав. кафедрой _____