

**Автономная образовательная некоммерческая организация
высшего образования
«Институт Бизнеса и Информационных Систем»
(АОНО ВО «ИБИС»)**

Факультет Бизнеса и информационных систем
Кафедра Информационных технологий



**РАБОЧАЯ ПРОГРАММА
И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
дисциплины**

Б1.О.15 «Основы программирования и алгоритмизации»

Уровень образования:	<u>Высшее образование – бакалавриат</u>
Направление подготовки:	<u>09.03.02 Информационные системы и технологии</u>
Направленность (профиль):	<u>Информационные системы и сетевые технологии</u>
Форма обучения:	<u>Очная, заочная</u>
Составитель:	<u>к.ф.-м.н. Моисеев С.И.</u>

Воронеж 2023 г.

Разработчик рабочей программы дисциплины: к.ф.-м.н. Моисеев Сергей Игоревич

Рабочая программа дисциплины рассмотрена и утверждена на заседаниях:
кафедры «Информационных технологий», протокол №2 от «25» апреля 2023 года.

Ученого совета АОНО «Институт Бизнеса и Информационных Систем», протокол
№ 3 от «11» мая 2023 года.

1. Цели и задачи учебной дисциплины

Цель освоения дисциплины «Основы программирования и алгоритмизации»: является формирование у обучающихся твердых теоретических знаний и практических навыков по составлению программ с использованием современной методологии программирования, по подготовке и представлению подпрограмм, различным пользователям для выработки, обоснования и принятия решений в области разработки современных программных продуктов.

Задачи дисциплины:

- ознакомление с принципами, базовыми концепциями технологий программирования, выступающими как составная часть технологии разработки объектов профессиональной деятельности в информационных системах экономического, управленческого, производственного, научного назначения;
- формирование и развитие компетенций, знаний, практических навыков и умений, обеспечивающих разработку средств реализации информационных технологий (в первую очередь информационных, алгоритмических и программных);
- практическое освоение интегрированной среды изучаемого алгоритмического языка высокого уровня;
- изучение основных этапов и принципов создания программного продукта, конструктивных компонентов и структуры компьютерных программ;
- знакомство с основными структурами данных, способами их представления и обработки;
- изучение методов обработки исключений, ошибок и отладок.

2. Место дисциплины в структуре ОПОП ВО

Дисциплина «Основы программирования и алгоритмизации» относится к обязательной части дисциплин, базируется на знаниях, полученных обучающимися в процессе освоения дисциплины «Информатика». Знания и умения, полученные в ходе изучения дисциплины «Основы программирования и алгоритмизации», будут необходимы при изучении дисциплин: Объектно-ориентированное программирование, Web-программирование в информационных системах.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенные с установленными в ОП ВО индикаторами достижения компетенций

Задача профессиональной деятельности	Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине
Применение современных информационно-коммуникационных технологий процессе осуществления профессиональных функций	ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности	ОПК-2.1 Знать: современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.	Знает современные средства разработки и анализа программного обеспечения на языках высокого уровня
		ОПК-2.2 Уметь:	

		<p>выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности.</p>	<p>инструментальными средами разработки программного обеспечения</p>
		<p>ОПК-2.3 Иметь навыки: применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности.</p>	<p>Имеет навыки работы с инструментальной средой создания и отладки программного кода</p>
<p>Разработка и программная реализация прикладных алгоритмов</p>	<p>ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий</p>	<p>ОПК-6.1 Знать: методы алгоритмизации, языки и технологии программирования, пригодные для практического применения в области информационных систем и технологий.</p>	<p>Знает основные структуры данных в языках программирования, методы программирования и разработки эффективных алгоритмов решения прикладных задач</p>
		<p>ОПК-6.2 Уметь: применять методы алгоритмизации, языки и технологии программирования при решении профессиональных задач в области информационных систем и технологий.</p>	<p>Умеет выбирать рациональные структуры данных и алгоритмы их обработки, обеспечивающие эффективную программную реализацию;</p>
		<p>ОПК-6.3 Иметь навыки: программирования, отладки и тестирования прототипов программно-технических комплексов задач.</p>	<p>Имеет навыками составления, тестирования и отладки программы на языке Си</p>

4. Объем и структура дисциплины

Общая трудоемкость дисциплины составляет 5 з.е., 180 часов.

Вид учебной работы	Формы обучения					
	Очная			Заочная		
	Всего часов	из них в семестре		Всего часов	из них в семестре	
		1			1	
Общая трудоемкость дисциплины	180	180		180	180	
Контактная работа обучающихся с преподавателем, всего	72	72		22	22	
в том числе:						
Лекции	36	36		6	6	
Лабораторные работы						
Практические занятия	36	36		16	16	
Самостоятельная работа	72	72		149	149	
Промежуточная аттестация (подготовка и сдача)	36	36		9	9	
Курсовая работа/проект						
Контрольная работа						
Промежуточная аттестация: экзамен/зачет/зачет с оценкой	Экзамен	Экзамен		Экзамен	Экзамен	

5. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

Содержание тем дисциплины, структурированное по темам с указанием дидактического материала по каждой изучаемой теме

№ п/п	Наименование темы	Содержание темы
1	Раздел 1. Основные понятия алгоритмизации и программирования	Основные понятия алгоритмизации и программирования. Проектирование алгоритмов (основные принципы и методологические подходы).
2	Раздел 2. Принципы структурной алгоритмизации	Основные классы программных алгоритмов. Линейные, ветвящиеся, циклические процессы. Основные принципы проектирование алгоритмов. Принципы структурного и модульного программирования.
3	Раздел 3. Язык PascalABC.NET. Синтаксис и семантика языка	Язык PascalABC.NET. Синтаксис и семантика языка. Базовые элементы языка Паскаль: алфавит; зарезервированные слова; идентификаторы; константы и переменные; типы данных; выражения, операции, операнды.
4	Раздел 4. Структура программы на PascalABC.NET	Структура программы на PascalABC.NET: общие сведения; описание меток; определение констант; описание переменных; определение типов данных; описание процедур и функций.
5	Раздел 5. Операторы языка PascalABC.NET	Операторы языка PascalABC.NET: общие сведения; простые операторы; сложные операторы.
6	Раздел 6. Основы типизации	Основы типизации и структуризации данных. Стро-

	зации и структуризации данных	ки, массивы, множества, записи. Сортировка и поиск.
7	Раздел 7. Процедуры и функции	Процедуры и функции: общие сведения; стандартные процедуры и функции; процедуры и функции пользователя. Рекурсия и итерация.
8	Раздел 8. Организация ввода-вывода	Ввод данных с клавиатуры. Вывод информации на дисплей. Файлы. Текстовые файлы.
9	Раздел 9. Документирование программных средств	Документирование программных средств. ЕСПД.
10	Раздел 10. Объектно-ориентированный подход к разработке программ	Объектно-ориентированный подход к разработке программ. Развитие технологии программирования.
11	Раздел 11. Основные понятия ООП	Основные понятия объектно-ориентированного программирования. Объект. Состояние и поведение. Операции над объектами. Основные понятия ООП: инкапсуляция, наследование, полиморфизм.
12	Раздел 12. Среда программирования PascalABC.NET. Основы визуального программирования	Среда программирования PascalABC.NET. Основы визуального программирования.
13	Раздел 13. Язык программирования Object Pascal	Язык программирования Object Pascal. Операторы. Процедуры и функции. Строки. Структурированные типы. Тип variant. Динамическая память и указатели.
14	Раздел 14. PascalABC.NET: структура проекта; модули, формы	Структура программы. Модули. Формы.
15	Раздел 15. Понятия объекта, класса	Понятия объекта, класса.
16	Тема 16. Виды классов. Наследование. Разработка Windows-приложений	Основные элементы класса: поля, методы, указатель this, конструкторы, деструкторы, операции. Дружественные функции и классы. Указатели на элементы классов. Простое и множественное наследование классов. Виртуальные методы. Абстрактные классы. Создание и использование шаблонов классов.
17	Тема 17. Интерфейсы и структуры.	Пользовательские и стандартные интерфейсы. Структуры. Интерфейсы. Принципы использования интерфейсов
18	Тема 18. Связанные списки на основе рекурсивных данных	Стек и очередь. Понятие рекурсивных данных и однонаправленные списки. Процедуры для работы со стеками. Процедуры для работы с односвязными очередями. Работа с двухсвязными очередями. Процедуры для работы с двусвязными очередями

Тематический план (очная форма обучения)

№ п/п	Наименование тем	Всего часов по учебному плану	Контактная работа с преподавателем:					Самостоятельная работа
			Всего часов	Лекции	Занятия семинарского типа			
					Семинарские занятия	Практические занятия	Другие виды занятий	
1 семестр								
1	Тема 1. Основные понятия алгоритмизации и программирования	6	2	2				4
2	Тема 2. Принципы структурной алгоритмизации	6	2	2				4
3	Тема 3. Язык PascalABC.NET. Синтаксис и семантика языка	6	2	2				4
4	Тема 4. Структура программы на PascalABC.NET	6	2	2				4
5	Тема 5. Операторы языка PascalABC.NET	8	4	2		2		4
6	Тема 6. Основы типизации и структуризации данных	10	6	2		4		4
7	Тема 7. Процедуры и функции	10	6	2		4		4
8	Тема 8. Организация ввода-вывода	10	6	2		4		4
9	Тема 9. Документирование программных средств	10	6	2		4		4
	Тема 10. Объектно-ориентированный подход к разработке программ	8	4	2		2		4
	Тема 11. Основные понятия ООП	8	4	2		2		4
	Тема 12. Среда программирования PascalABC.NET. Основы визуального программирования	8	4	2		2		4
	Тема 13. Язык программирования Object Pascal	8	4	2		2		4
	Тема 14. PascalABC.NET: структура проекта; модули, формы	8	4	2		2		4
	Тема 15. Понятия объекта, класса	8	4	2		2		4
	Тема 16. Виды классов. Наследование. Разработка Windows-приложений	8	4	2		2		4
	Тема 17. Интерфейсы и структуры.	8	4	2		2		4
	Тема 18. Связанные списки на основе рекурсивных данных	8	4	2		2		4
Всего за семестр:		144	72	36		36		72
Форма контроля: Экзамен		36						36
Итого за семестр:		180	72	36		36		108

Тематический план (заочная форма обучения)

№ п/п	Наименование тем	Всего часов по учебному плану	Контактная работа с преподавателем:				Самостоятельная работа	
			Всего часов	Лекции	Занятия семинарского типа			
					Семинарские занятия	Практические занятия		Другие виды занятий
1 семестр								
1	Тема 1. Основные понятия алгоритмизации и программирования	8					8	
2	Тема 2. Принципы структурной алгоритмизации	10	2	2			8	
3	Тема 3. Язык PascalABC.NET. Синтаксис и семантика языка	10	2			2	8	
4	Тема 4. Структура программы на PascalABC.NET	8					8	
5	Тема 5. Операторы языка PascalABC.NET	12	4	2		2	8	
6	Тема 6. Основы типизации и структуризации данных	10	2			2	8	
7	Тема 7. Процедуры и функции	8					8	
8	Тема 8. Организация ввода-вывода	10	2	2			8	
9	Тема 9. Документирование программных средств	10	2			2	8	
	Тема 10. Объектно-ориентированный подход к разработке программ	8					8	
	Тема 11. Основные понятия ООП	10	2			2	8	
	Тема 12. Среда программирования PascalABC.NET. Основы визуального программирования	8					8	
	Тема 13. Язык программирования Object Pascal	10	2			2	8	
	Тема 14. PascalABC.NET: структура проекта; модули, формы	8					8	
	Тема 15. Понятия объекта, класса	10	2			2	8	
	Тема 16. Виды классов. Наследование. Разработка Windows-приложений	10					10	
	Тема 17. Интерфейсы и структуры.	10					10	
	Тема 18. Связанные списки на основе рекурсивных данных	11	2			2	9	
	Всего за семестр:	171	22	6		16	149	
	Форма контроля: Экзамен	9					9	
	Итого за семестр:	180	22	6		16	108	

6. Самостоятельная работа обучающихся в ходе освоения дисциплины

№ п/п	Вид самостоятельной работы	Наименование работы и содержание
1	Освоение учебного материала по конспекту лекций и дополнительной литературе	Доработать конспект, желательно в тот же день. Прочитать записи, восстановить текст в памяти, а также исправить описки, расшифровать не принятые ранее сокращения, заполнить пропущенные места, понять текст, вникнуть в его смысл. Изучить материал, используя рекомендуемую литературу, разрешая в ходе чтения, возникшие ранее затруднения, находя ответы на вопросы, а также дополняя и исправляя свои записи. Записи должны быть наглядными, для чего следует применять различные способы выделений. Подготовленный конспект и рекомендуемая литература используются при подготовке к практическому занятию.
2	Подготовка к практическим занятиям	Подготовка к практическому занятию включает следующие элементы самостоятельной деятельности: четкое представление цели и задач его проведения; выделение навыков умственной, аналитической деятельности, которые станут результатом предстоящей работы. Выработка навыков осуществляется с помощью получения новой информации об изучаемых процессах и с помощью знания о том, в какой степени в данное время студент владеет методами исследовательской деятельности, которыми он станет пользоваться на практическом занятии.
3	Изучение основной и дополнительной литературы	Самостоятельная работа с учебниками и книгами (а также самостоятельное теоретическое исследование проблем, обозначенных преподавателем на лекциях) – это важнейшее условие познания. В самостоятельной работе рекомендуется прибегать к таким видам систематизированной записи прочитанного как аннотирование, тезирование, цитирование, конспектирование. Причем конспект аккумулирует в себе предыдущие виды записи, позволяет всесторонне охватить содержание книги, статьи. Поэтому умение составлять план, тезисы, делать выписки и другие записи определяет и технологию составления конспекта.
4	Подготовка к экзамену	Необходимо перечитать лекции, вспомнить то, что говорилось преподавателем на семинарах и практических занятиях, а также самостоятельно полученную информацию при подготовке к ним. важно сформировать целостное представление о содержании ответа на каждый вопрос, что предполагает знание разных научных трактовок сущности того или иного явления, процесса, умение раскрывать факторы, определяющие их противоречивость, знание имен ученых, изучавших обсуждаемую проблему. необходимо также привести информацию о материалах эмпирических исследований, что указывает на всестороннюю подготовку обучающегося к экзамену ответ, в котором присутствуют все указанные блоки информации, наверняка будет отмечен высокими баллами. для их получения требуется ответить и на дополнительные вопросы, если экзамен проходит в устной форме. Рекомендуется подготовку к экзамену осуществлять в два

		этапа. На первом, в течение 2–3 дней, подбирается из разных источников весь материал, необходимый для развернутых ответов на все вопросы. ответы можно записать в виде краткого конспекта. На втором этапе по памяти восстанавливается содержание того, что записано в ответах на каждый вопрос.
--	--	--

7. Фонд оценочных средств для текущей и промежуточной аттестации по дисциплине

Оценочные средства для проведения текущей и промежуточной аттестации

Код контролируемого индикатора освоения компетенции	Наименование оценочного средства для проведения текущей аттестации	Наименование оценочного средства для проведения промежуточной аттестации
ОПК-2.1, ОПК-2.2, ОПК-2.3 ОПК-6.1, ОПК-6.2, ОПК-6.3	Опрос, тестовые задания, практические работы	Экзамен

Критерии оценивания результата обучения по дисциплине и шкала оценивания

Код контролируемой компетенции	Критерии оценивания результата обучения по дисциплине и шкала оценивания			
	неудовлетворительно	удовлетворительно	хорошо	отлично
	Не зачтено	Зачтено		
ОПК-2	обучающийся обнаруживает незнание ответа на соответствующее задание, допускает принципиальные ошибки в формулировке определений и правил, в течение семестра не сформировал необходимых умений и навыков	обучающийся демонстрирует удовлетворительное, но не систематизированное владение принципами работы современных информационных технологий и программных средств, в том числе отечественного производства, и использование их при решения задач профессиональной деятельности	обучающийся демонстрирует достаточно полное, с небольшими неточностями, владение принципами работы современных информационных технологий и программных средств, в том числе отечественного производства, и использование их при решения задач профессиональной деятельности	обучающийся демонстрирует полное, систематизированное владение принципами работы современных информационных технологий и программных средств, в том числе отечественного производства, и использование их при решения задач профессиональной деятельности

ОПК-6	обучающийся обнаруживает незнание ответа на соответствующее задание, допускает принципиальные ошибки в формулировке определений и правил, в течение семестра не сформировал необходимых умений и навыков	обучающийся демонстрирует удовлетворительное, но не систематизированное владение способностями разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	обучающийся демонстрирует достаточно полное, с небольшими неточностями, владение способностями разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	обучающийся демонстрирует полное, систематизированное владение способностями разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий
-------	--	---	---	---

8. Ресурсное обеспечение учебной дисциплины

Основная литература:

1. Немцова, Т. И. Программирование на языке высокого уровня. Программирование на языке Object Pascal : учебное пособие / Т. И. Немцова, С. Ю. Голова, И. В. Абрамова ; под ред. Л. Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2020. — 496 с. — (Высшее образование: Бакалавриат). - ISBN 978-5-8199-0901-0. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1044632> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

2. Затонский, А. В. Программирование и основы алгоритмизации. Теоретические основы и примеры реализации численных методов: учебное пособие / А.В. Затонский, Н.В. Бильфельд. — 2-е изд. — Москва : РИОР : ИНФРА-М, 2020. — 167 с. — (Высшее образование). — DOI: <https://www.dx.doi.org/10.12737/20468>. - ISBN 978-5-369-01195-9. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1077389> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

Дополнительная литература:

1. Бедердинова, О. И. Программирование на языках высокого уровня : учеб. пособие / О.И. Бедердинова, Т.А. Минеева, Ю.А. Водовозова. — Москва : ИНФРА-М, 2019. — 159 с. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1044396> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

2. Кузин, А. В. Программирование на языке Си : учебное пособие / А. В. Кузин, Е. В. Чумакова. — Москва : ФОРУМ : ИНФРА-М, 2019. — 144 с. — (Высшее образование). - ISBN 978-5-00091-066-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1007488> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

3. Литвиненко, В. А. Программирование на C++ задач на графах: Учебное пособие / Литвиненко В.А. - Таганрог: Южный федеральный университет, 2016. - 83 с.: ISBN 978-5-9275-2311-5. - Текст : электронный. - URL: <https://znanium.com/catalog/product/997083> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

4. Гуриков, С. Р. Программирование в среде Lazarus : учеб. пособие / С.Р. Гуриков. — Москва : ФОРУМ : ИНФРА-М, 2019. — 336 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-555-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/961652> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

5. Кучунова, Е. В. Программирование. Процедурное программирование: Учебное пособие / Кучунова Е.В., Олейников Б.В., Чередниченко О.М. - Краснояр.:СФУ, 2016. - 92 с.: ISBN 978-5-7638-3555-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/978627> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

6. Комлев, Н. Ю. Объектно Ориентированное Программирование. Хорошая книга для Хороших Людей / Комлев Н.Ю. - Москва :СОЛОН-Пр., 2015. - 298 с.:ISBN 978-5-91359-138-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/884394> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

7. Воронцова, Е. А. Программирование на С++ с погружением: практические задания и примеры кода - Москва :НИЦ ИНФРА-М, 2016. - 80 с. ISBN 978-5-16-105159-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/563294> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

8. Кузин, А. В. Программирование на языке Си : учебное пособие / А.В. Кузин, Е.В. Чумакова. — Москва : ФОРУМ : ИНФРА-М, 2020. — 143 с. — (Среднее профессиональное образование). - ISBN 978-5-00091-556-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/961653> (дата обращения: 31.10.2021). – Режим доступа: по подписке.

Электронные ресурсы:

1. Интернет Университет Информационных технологий. [Электронный ресурс] : сайт. – Режим доступа: URL: <http://www.intuit.ru/>, свободный (дата обращения 30.09.2021)

2. Портал естественных наук. [Электронный ресурс] : сайт. – Режим доступа: URL: <http://e-science11.ru>, свободный (дата обращения 30.09.2021).

3. eLIBRARY.RU [Электронный ресурс] : научная электронная библиотека. – Режим доступа: URL: <http://elibrary.ru/defaultx.asp>, свободный (дата обращения: 30.09.2021).

4. Единое окно доступа к образовательным ресурсам [Электронный ресурс] : федеральный портал. – Режим доступа: URL: <http://window.edu.ru/>, свободный (дата обращения: 30.09.2021)

5. Сайт ixbt.com [Электронный ресурс] Полная оперативная и объективная информация о персональных компьютерах, их компонентах и периферийных устройствах. – Режим доступа: URL: www.ixbt.com свободный (дата обращения: 30.09.2021).

6. Сайт CITForum [Электронный ресурс] Библиотека технических материалов по информационным технологиям. – Режим доступа: URL: www.citforum.ru свободный (дата обращения: 30.09.2021).

9. Материально-техническое обеспечение дисциплины

Учебная аудитория № 315 -учебная аудитория для проведения занятий лекционного типа; -учебная аудитория для проведения занятий семинарского тип и практических занятий; -учебная аудитория групповых и индивидуальных консультаций; -учебная аудитория для проведения текущего контроля и промежу-	394026, Воронежская область, г. Воронеж, ул. Дружинников, д.8 Кабинет № 315
---	---

<p>точной аттестации.</p> <p>Оснащение оборудованием и техническими средствами обучения:</p> <ul style="list-style-type: none"> - комплект учебной мебели для обучающихся; - рабочее место преподавателя; - доска меловая; - стационарное видеопроекционное оборудование для мультимедиа презентации, средства звуковоспроизведения (проектор, персональный компьютер, колонки, Web-камера). <p>Лицензионное программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - MS Windows 10; - Microsoft Office Standard 2007. <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - Kaspersky EndPoint Security для Windows. <p>Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации.</p>	<p>(3 этаж № 12)</p>
<p>Учебная аудитория № 314</p> <ul style="list-style-type: none"> -учебная аудитория для проведения занятий лекционного типа; -учебная аудитория для проведения занятий семинарского тип и практических занятий; -учебная аудитория групповых и индивидуальных консультаций; -учебная аудитория для проведения текущего контроля и промежуточной аттестации; -учебная аудитория для курсового проектирования (выполнения курсовых работ); -помещение для самостоятельной работы обучающихся; - компьютерный класс. <p>Оснащение оборудованием и техническими средствами обучения:</p> <ul style="list-style-type: none"> -автоматизированное рабочее место обучающегося; -автоматизированное рабочее место преподавателя; -доска двусторонняя (маркерно - меловая); -наушники; -принтер; -телевизор. <p>Лицензионное программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - MS Windows 8.1 Корпоративная; - Microsoft Office Standard 2007; - iSpring suite 8; - MS Visio; - MS Access 2016; - MS Project; - Microsoft SQL Server 2014; - Visual Studio 2017. <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - Kaspersky EndPoint Security для Windows; <p>-1С: Предприятия 8. Комплект для обучения в высших и средних учебных заведениях.</p> <p>Свободно распространяемое программное обеспечение иностранного производства:</p> <ul style="list-style-type: none"> - PascalABC.NET; 	<p>394026, Воронежская область, г. Воронеж, ул. Дружинников, д.8 Кабинет № 314 (3 этаж № 48)</p>

<ul style="list-style-type: none"> - FreePascal IDE; - Eclipse; - IntelliJ IDEA; - GIMP; - Blender; - Firefox; - Vuze; - FileZilla; - Denver, Maxima + WxMaxima; - iTest; - Inkscape; - QCad. <p>Информационная справочная правовая система «Консультант Плюс».</p> <p>Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации.</p>	
<p>Учебная аудитория № 318</p> <ul style="list-style-type: none"> -учебная аудитория для проведения занятий лекционного типа; -учебная аудитория для проведения занятий семинарского тип и практических занятий; -учебная аудитория групповых и индивидуальных консультаций; -учебная аудитория для проведения текущего контроля и промежуточной аттестации; -учебная аудитория для курсового проектирования (выполнения курсовых работ); -компьютерный класс; -помещение для самостоятельной работы обучающихся. <p>Оснащение оборудованием и техническими средствами обучения:</p> <ul style="list-style-type: none"> -автоматизированное рабочее место обучающегося; -автоматизированное рабочее место преподавателя; -доска двусторонняя (маркерно-меловая). - переносное видеопроекторное оборудование для мультимедиа презентации (ноутбук, проектор, экран, колонки). <p>Лицензионное программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - MS Windows 7; - Microsoft Office Standard 2007; - MS Visio 2007; - MS Project 2010; - Microsoft SQL Server 2012; - Microsoft Visual Studio. <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - Kaspersky EndPoint Security для Windows; - Автоматизированная банковская система «Управление кредитной организацией» для ВУЗов. <p>Свободно распространяемое программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - PascalABC.NET; - FreePascal IDE; - GIMP; - Blender; 	<p>394026, Воронежская область, г. Воронеж, ул. Дружинников, д.8 Кабинет № 318 (3 этаж № 50)</p>

<ul style="list-style-type: none"> - Firefox; - Vuze; - FileZilla; - Denver; - Maxima + WxMaxima; - iTest; - Inkscape; - QCad; <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - программа Фоторобот. <p>Российская информационная справочная правовая система «Консультант Плюс».</p> <p>Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации.</p>	
<p>Помещение для самостоятельной работы обучающихся № 102</p> <ul style="list-style-type: none"> - помещение для самостоятельной работы обучающихся с доступом к сети «Интернет» и электронной информационно-образовательной среде организации; - читальный зал библиотеки - учебная аудитория для курсового проектирования (выполнения курсовых работ); - учебная аудитория для выполнения и защиты выпускной квалификационной работы. <p>Оснащение оборудованием и техническими средствами обучения:</p> <ul style="list-style-type: none"> - автоматизированное рабочее место обучающегося; - ноутбуки; - телевизор; - столы для чтения; - стулья; - шкафы для документов; - стол офисный; - стеллажи для книг; - стойка выдачи литературы; - тумба напольная; - информационная стойка. <p>Лицензионное программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - MS Windows 7 pro; - Microsoft Office Standard 2007; - MS Access 2016. <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - Kaspersky EndPoint Security для Windows; Свободно распространяемое программное обеспечение: - 7-Zip; - Интернет цензор. <p>Российская информационная справочная правовая система «Консультант Плюс».</p> <p>Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации</p>	<p>394026, Воронежская область, г. Воронеж, ул. Дружинников, д.8 Кабинет № 102 (1 этаж № 84)</p>
<p>Учебная аудитория № 314</p>	<p>394026, Воронеж-</p>

<p>- помещение для самостоятельной работы обучающихся с доступом к сети «Интернет» и электронной информационно-образовательной среде организации;</p> <p>-учебная аудитория для курсового проектирования (выполнения курсовых работ);</p> <p>-учебная аудитория для выполнения выпускной квалификационной работы;</p> <p>- компьютерный класс.</p> <p>Оснащение оборудованием и техническими средствами обучения:</p> <p>-автоматизированное рабочее место обучающегося; - автоматизированное рабочее место преподавателя; -доска двусторонняя (маркерно - меловая);</p> <p>-наушники;</p> <p>-принтер;</p> <p>-телевизор.</p> <p>Лицензионное программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - MS Windows 8.1 Корпоративная; - Microsoft Office Standard 2007; - iSpring suite 8; - MS Visio; - MS Access 2016; - MS Project; - Microsoft SQL Server 2014; - Visual Studio 2017. <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - Kaspersky EndPoint Security для Windows; -1С: Предприятия 8. Комплект для обучения в высших и средних учебных заведениях. <p>Свободно распространяемое программное обеспечение иностранного производства:</p> <ul style="list-style-type: none"> - PascalABC.NET; - FreePascal IDE; - Eclipse; - IntelliJ IDEA; - GIMP; - Blender; - Firefox; - Vuze; - FileZilla; - Denver, Maxima + WxMaxima; - iTest; - Inkscape; - QCad. <p>Информационная справочная правовая система «Консультант Плюс».</p> <p>Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации.</p>	<p>ская область, г. Воронеж, ул. Дружинников, д.8 Кабинет № 314 (3 этаж № 48)</p>
<p>Учебная аудитория № 318</p> <p>- помещение для самостоятельной работы обучающихся с доступом к сети «Интернет» и электронной информационно-</p>	<p>394026, Воронеж-ская область, г. Воронеж,</p>

<p>образовательной среде организации;</p> <ul style="list-style-type: none"> -учебная аудитория для курсового проектирования (выполнения курсовых работ); -учебная аудитория для выполнения выпускной квалификационной работы; <p>Оснащение оборудованием и техническими средствами обучения:</p> <ul style="list-style-type: none"> -автоматизированное рабочее место обучающегося; - автоматизированное рабочее место преподавателя; -доска двусторонняя (маркерно-меловая); - переносное видеопроекторное оборудование для мультимедиа презентации (ноутбук, проектор, экран, колонки). <p>Лицензионное программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - MS Windows 7; - Microsoft Office Standard 2007; - MS Visio 2007; - MS Project 2010; - Microsoft SQL Server 2012; - Microsoft Visual Studio. <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - Kaspersky EndPoint Security для Windows; -Автоматизированная банковская система «Управление кредитной организацией» для ВУЗов. <p>Свободно распространяемое программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - PascalABC.NET; - FreePascal IDE; - GIMP; - Blender; - Firefox; - Vuze; - FileZilla; - Denver; - Maxima + WxMaxima; - iTest; - Inkscape; - QCad; <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - программа Фоторобот. <p>Российская информационная справочная правовая система «Консультант Плюс».</p> <p>Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации.</p>	<p>ул. Дружинников, д.8 Кабинет № 318 (3 этаж № 50)</p>
<p>Учебная аудитория № 313</p> <ul style="list-style-type: none"> - помещение для самостоятельной работы обучающихся с доступом к сети «Интернет» и электронной информационно-образовательной среде организации; -учебная аудитория для курсового проектирования (выполнения курсовых работ); -учебная аудитория для выполнения выпускной квалификационной работы; 	<p>394036, город Воронеж, ул. Карла Маркса, д.67 Кабинет № 313 (3 этаж № 62)</p>

<p>- компьютерный класс.</p> <p>Оснащение оборудованием и техническими средствами обучения:</p> <ul style="list-style-type: none"> - автоматизированное рабочее место обучающегося; - автоматизированное рабочее место преподавателя; - доска маркерная; - стационарное видеопроекционное оборудование для мультимедиа презентации, средства звуковоспроизведения (экран, проектор, колонки). <p>Лицензионное программное обеспечение:</p> <p>1) иностранного производства:</p> <ul style="list-style-type: none"> - MS Windows 10; - Microsoft Office Standard 2007; - MS Visio; - MS Access 2016; - MS Project; - Microsoft SQL Server 2019; - Visual Studio 2010; <p>2) отечественного производства:</p> <ul style="list-style-type: none"> - Kaspersky EndPoint Security для Windows. <p>Свободно распространяемое программное обеспечение иностранного производства:</p> <ul style="list-style-type: none"> - PascalABC.NET; - FreePascal IDE; - Eclipse; - IntelliJ IDEA; - GIMP; - Blender; - Firefox; - Vuze; - FileZilla; - Denver; - Maxima + WxMaxima, iTest; - Inkscape; - QCad. <p>Российская информационная справочная правовая система «Консультант Плюс».</p> <p>Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации.</p>	
<p>Учебная аудитория № 314</p> <ul style="list-style-type: none"> - помещение для самостоятельной работы обучающихся с доступом к сети «Интернет» и электронной информационно-образовательной среде организации; - учебная аудитория для курсового проектирования (выполнения курсовых работ); - учебная аудитория для выполнения выпускной квалификационной работы; <p>- компьютерный класс.</p> <p>Оснащение оборудованием и техническими средствами обучения:</p> <ul style="list-style-type: none"> - автоматизированное рабочее место обучающегося; - автоматизированное рабочее место преподавателя; - доска двусторонняя (маркерно - меловая); 	<p>394036, город Воронеж, ул. Карла Маркса, д.67 Кабинет № 314 (3 этаж № 61)</p>

<p>-наушники; -принтер; -телевизор. Лицензионное программное обеспечение: 1) иностранного производства: - MS Windows 8.1 Корпоративная; - Microsoft Office Standard 2007; - iSpring suite 8; - MS Visio; - MS Access 2016; - MS Project; - Microsoft SQL Server 2014; - Visual Studio 2017. 2) отечественного производства: - Kaspersky EndPoint Security для Windows; -1С: Предприятия 8. Комплект для обучения в высших и средних учебных заведениях. Свободно распространяемое программное обеспечение иностранного производства: - PascalABC.NET; - FreePascal IDE; - Eclipse; - IntelliJ IDEA; - GIMP; - Blender; - Firefox; - Vuze; - FileZilla; - Denver, Maxima + WxMaxima; - iTest; - Inkscape; - Qcad. Информационная справочная правовая система «Консультант Плюс». Доступ к информационно-телекоммуникационной сети «Интернет» и к электронной информационно-образовательной среде организации.</p>	
---	--

10. Оценочные материалы для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине

10.1 Материалы для текущего контроля освоения дисциплины

Тема 1. Понятие алгоритма и алгоритмического языка

Средства текущего контроля для проведения опроса:

- 1.Алгоритм, его свойства, способы представления.
- 2.Программы на языках высокого уровня.
- 3.Постановка задачи и спецификация программы.
- 4.Основные этапы решения задач на ЭВМ.
5. Критерии качества программ; диалоговые программы.
6. Жизненный цикл программ.

7. Каковы возможные подходы к определению понятия алгоритм?
8. Кто (что) может быть исполнителем алгоритма?
9. В чем особенности графического способа представления алгоритмов?
10. Каковы основные алгоритмические структуры?
11. Чем определяются свойства алгоритмов «дискретность», «определенность», «понятность», «результативность», «массовость»?
12. Что такое алгоритмический язык?

Тема 2. Принципы структурной алгоритмизации

Средства текущего контроля для проведения опроса:

1. Определение типа данных.
2. Статический и динамический контроль типов и классификация языков в соответствии с ним.
3. Слабая, сильная и строгая типизации в статически типизируемых языках.
4. Конструкции структурного программирования и теорема о структурировании.
5. Абстракция управления.

Тема 3 Типы данных, операции и операторы языка Паскаль

Средства текущего контроля для проведения опроса:

1. Понятие переменной.
2. Команда присваивания. Формат, примеры.
3. Общие сведения о вводе-выводе данных.
4. Процедуры ввода данных. Read. Формат, примеры.
5. Процедура ReadLn. Формат, примеры.
6. Процедуры вывода данных. Write, WriteLn. Форматы, примеры.

Тема 4 Язык PascalABC.NET. Синтаксис и семантика языка

Средства текущего контроля для проведения опроса:

1. Лексическая структура языка. Структура паскаль- программ, разделы и их назначение.
2. Понятия типа, иерархия типа. Целые типы, литерный тип, вещественные типы. Генератор случайных чисел.
3. Булевский тип. Логические операции, логические функции.
4. Выражения, правила их записи и выполнения, классификация выражения по типу результата.
5. Ввод информации, вывод информации на экран. Форматный вывод.
6. Программирование ветвлений. Понятие цикла, циклы с заданным числом повторений и итерационные. Специальный выход из цикла.
7. Вложенные циклы, их организация с помощью всех трех операторов. Правила использования.
8. Перечислимый тип. Правила использования, действия с данными указанного типа
9. Ограниченный тип, области его применения. Функции low, high. Приведение типа переменной.
10. Одномерные, двумерные массивы, ввод/вывод, использование индексов нечислового типа. Ошибки в работе с регулярными типами, режимы компиляции. \$R+ и \$R-..
11. Строковый тип. Процедуры и функции для обработки строк. (пример программы обработки текста с использованием строк).
12. Множества, действия с множествами.
13. Понятие структуры данных запись. Иерархические записи, массивы записи, оператор присоединения. Записи с вариантами, действия с ними, размещение в памяти.

Тема 5 Операторы языка PascalABC.NET

Средства текущего контроля для проведения опроса:

1. Процедуры. Описание и вызов.
2. Формальные и фактические параметры, их взаимодействие.
3. Параметры-переменные, параметры-значения, параметры-константы.
4. Параметры сложных типов.
5. Бестиповые параметры. Открытые параметры и строки.
6. Функции. Описание, вызов.
7. Область действия имен.
8. Формальные и фактические параметры.
9. Локальные и глобальные идентификаторы.
10. Процедурный и функциональный типы. Правила использования.
11. Рекурсивные алгоритмы.
12. Порядок отведения памяти под локальные параметры. Пример программы.
13. Синтаксические анализаторы.
14. Элементы металингвистических формул. Пример программы.

Практическая работа № 1 Способы записи алгоритмов. Составление блок-схем

Цель работы: научиться работать с алгоритмами и блок-схемами. Научиться составлять алгоритмы различных структур.

Вопросы по практической работе 1

1. Что такое блок-схема?
2. Какие типы блоков бывают?
3. Какие блоки используются при реализации линейного, разветвляющегося, циклического алгоритмов?
4. Можно ли составить разные варианты блок-схем для одной и той же задачи?
5. Какие виды циклического алгоритма бывают?
6. Какие пункты должны присутствовать в любом цикле?
7. Что такое выполнение блок-схемы?
8. Для чего следует выполнять блок-схему?

Тема 6 Основы типизации и структуризации данных

Средства текущего контроля для проведения опроса:

1. Абстракция модульности.
2. Модульное программирование.
3. Принцип скрытия информации.
4. Аксиома модульности.
5. Принцип сборочного программирования.
6. Определение модуля.
7. Модули в языках программирования.
8. Процедурная абстракция.
9. Основные характеристики программного модуля.
10. Гипотеза о глобальных данных.
11. Размер модуля.
12. Сцепление модулей.
13. Связность модуля.
14. Методы разработки структуры программы.

Практическая работа № 2 Использование стандартных типов, задание типов определяемых пользователем при разработке программ

Цель работы: ознакомиться со структурой программы, научиться работать со всеми типами данных, операторами.

Вопросы по практической работе 2

1. Определение типа данных.
3. Статический и динамический контроль типов и классификация языков в соответствии с ним.
2. Слабая, сильная и строгая типизации в статически типизируемых языках.
3. Конструкции структурного программирования и теорема о структурировании.
4. Абстракция управления.

Тема 7 Процедуры и функции

Средства текущего контроля для проведения опроса:

1. Структура функции, определенной пользователем.
2. Как осуществляется вызов функции из основной программы?
3. Каким образом функция передает результат в основную программу?
4. Описание процедуры. Общая структура.
5. Как осуществляется вызов процедуры из основной программы?
6. Каким образом процедура передает результат в основную программу?
7. Что такое формальные и фактические параметры?
8. Какая взаимосвязь существует между формальными и фактическими параметрами?
9. Какие виды параметров могут быть указаны при описании процедуры или функции в ее заголовке?
10. Для чего используются параметры-значения?
11. Для чего используются параметры-переменные?
12. Процедуры и функции модуля CRT.TPU.
13. Процедуры и функции модуля STRINGS.TPU.
14. Процедуры и функции модуля DOS.TPU.

Практическая работа № 3 Разработка и реализация программ с использованием стандартных и определенных пользователем типов данных

Цель работы: Приобретение навыков разработки программ с использованием процедур и функций пользователя

Вопросы по практической работе 3

1. Что называется подпрограммой?
2. Какова структура программы с подпрограммами?
3. Какова структура процедур?
4. Какова структура подпрограммы-функции?

Тема 8 Организация ввода-вывода

Средства текущего контроля для проведения опроса:

1. Какие процедуры служат в Паскале для выполнения операций ввода-вывода?
2. Напишите оператор ввода переменной K с клавиатуры.
3. Для каких целей служит оператор присваивания.
4. Чем отличаются операторы ввода Read и Readln?
5. Для каких целей служит оператор Write.
6. Чем отличаются операторы вывода Write и Writeln?
7. Для чего в процедурах вывода определяется ширина поля вывода?

8. Какие обозначения используются в форматах вывода?
9. Файлы. Файловые типы.
10. Понятие прямого и последовательного методов доступа.
11. Файловая система Турбо Паскаля: понятие логического и физического файлов.

Текстовые файлы.

12. Процедуры и функции для организации прямого метода доступа.
13. Алгоритмы формирования, просмотра и корректировки текстового и типизированного файлов.

Практическая работа № 4 Разработка программ линейной, разветвляющейся, циклической структуры. Составление подпрограмм – процедур, подпрограмм – функций

Цель работы: ознакомиться со структурой программы, научиться программировать простейшие структуры

Вопросы по практической работе 4

1. В чем состоит различие и сходство подпрограмм-процедур и подпрограмм-функций?
2. Как обратиться к процедурам и функциям?
3. Что называется параметром и каково его назначение?
4. Назначение формальных и фактических параметров и их взаимосвязь.
5. Опишите последовательность событий при вызове процедур или функций.
6. Для чего используется пошаговый режим с заходом в процедуры и как его осуществить?
7. Каковы отличия параметров-значений от параметров-переменных, особенности их описания и применения?
8. Чем отличаются локальные и глобальные параметры? Какова область их действия?
9. Какой алгоритм является алгоритмом циклической структуры?
10. Типы циклов в языке.
11. Какой цикл выполнится хотя бы один раз? А какой - ни разу?

Тема 9 Документирование программных средств

Средства текущего контроля для проведения опроса:

1. Ссылочные типы: стандартный и пользовательский. Действия с указателями.
2. Динамические структуры данных. Схема распределения памяти при выполнении программ.
3. Действия с динамическими переменными: создание и уничтожение. Динамические массивы и массивы указателей.
4. Функции для определения свободной динамической памяти.
5. Линейные списки.
6. Формирование, просмотр, удаление и включение элемента.
7. Примеры программ.
8. Стек. Формирование, добавление, исключение элемента из стека.
9. Примеры программ.
10. Очереди. Включение элемента в очередь и удаление.
11. Двухнаправленные списки.
12. Создание списка, удаление и вставка элемента.

Практическая работа № 5 Реализация программ использующих основные структуры программирования

Цель работы: ознакомиться с основными структурами программирования, научиться составлять программы с разветвляющимися и циклическими структурами

Вопросы по практической работе 5

1. Какой алгоритм является алгоритмом разветвляющейся структуры?
2. Что такое условный оператор? Оператор выбора?
3. Полная и сокращенная формы условного оператора.
4. Что такое составной оператор? Формат его записи.
5. В каких случаях применяется полная форма условного оператора?
6. Что такое метка? Описывается ли метка в программе?
7. Что такое оператор безусловного перехода?
8. В каких случаях нельзя обойтись без операторов безусловного перехода?
9. В каких случаях надо использовать CASE вместо IF?

Тема 10 Объектно-ориентированный подход к разработке программ

Средства текущего контроля для проведения опроса:

1. Перечислите классические типы. Опишите механизмы для создания новых типов.
2. В чем смысл декомпозиции при составлении программ?
3. В чем различие между видами и методами (способами) абстракции?
4. Дайте определение классам операций абстракции данных. Объясните смысл полноты класса операций.
5. Дайте понятие класса в ООП. Опишите отношение "объект - класс".
6. Сформулируйте понятие полиморфизма. Проявления полиморфизма.
7. Опишите механизм наследования классов.
8. Что такое модификатор доступа и наследования?

Практическая работа № 6 Разработка программ с переменными типа запись.

Работа с файлами данных

Цель работы: Научиться составлять алгоритмы различных структур. Познакомиться с неоднородными структурами данных – записи, файлы.

Вопросы по практической работе 6

1. Тип запись. Объявление.
2. Принципы обработки.
3. Определение файла.
4. Классификация файлов.
5. Процедуры и функции для обработки типизированных файлов.

Тема 11 Основные понятия ООП

Средства текущего контроля для проведения опроса:

1. Что такое структурное программирование, его отличительные особенности, достоинства и недостатки. Область применения технологии структурного программирования.
2. Нисходящее проектирование программы. Технология модульного программирования, ее отличительные особенности, достоинства и недостатки. Область применения технологии модульного программирования. Правила декомпозиции решаемой задачи (правило семи). Актуальна ли технология модульного программирования для технологии объектно-ориентированного программирования?
3. Технология объектно-ориентированного программирования (ООП). Основные принципы ООП и их краткая характеристика. Достоинства и недостатки ООП. Как можно использовать модульную технологию программирования в ООП?
4. Технология объектно-ориентированного программирования (ООП). Инкапсуляция (класс), ее особенности и достоинства. Как можно ограничивать доступ к членам класса? Как получить доступ к закрытым полям объекта-класса из произвольной программной среды? Приведите пример объявления какого-либо класса.

5. Технология объектно-ориентированного программирования (ООП). Класс. Что такое конструктор? Виды конструкторов и их краткая характеристика. Особенности оформления конструкторов. Области применения конструкторов. Всегда ли нужен конструктор? Каким способом вызывается конструктор и в какой момент времени?

6. Технология объектно-ориентированного программирования (ООП). Класс. Что такое конструктор копирования? Всегда ли нужен конструктор копирования? Каким способом вызывается конструктор копирования и в какой момент времени?

7. Технология объектно-ориентированного программирования (ООП).

8. Класс. Что такое деструктор? Особенности оформления деструктора.

9. Всегда ли нужен деструктор? Какими способами можно вызывать деструктор?

Практическая работа № 7 Создание программ с переменными типа запись

Цель работы: научиться составлять алгоритмы структуры типа запись

Вопросы по практической работе 7

1. Как описываются переменные типа запись?

2. В каких случаях целесообразно использовать переменные типа запись?

3. Из каких компонентов состоит переменная типа запись?

4. Каково назначение оператора WITH?

5. Какие операции допустимы над полями записи?

6. В чем отличие записей от массивов?

Тема 12 Среда программирования PascalABC.NET. Основы визуального программирования

Средства текущего контроля для проведения опроса:

1. Как запустить среду PascalABC.NET?

2. Как создать новую страницу? Как открыть текст ранее созданной программы?

3. Как закрыть текущую страницу с текстом программы? Можно ли закрыть все неактивные страницы?

4. Пусть у вас открыто несколько программ. Как сделать активной какую-либо из них?

5. Как исполнить программу, размещенную на какой-либо странице?

6. Какими словами ограничивается тело программы на языке Паскаль?

7. Какую команду (процедуру) следует использовать в программе для вывода текста?

Практическая работа № 8 Манипулирование адресами области памяти с помощью указателей

Цель работы: изучить особенности работы с динамическими переменными, работы с оперативной памятью, познакомиться со стандартными процедурами и функциями для работы с динамическими переменными

Вопросы по практической работе 8

1. Дайте понятие указателя. Для каких целей он служит?

2. Поясните следующие понятия: установка указателя, тип указателя.

3. Какие действия выполняют операции * и & ?

4. Какие действия необходимо выполнить, чтобы обратиться к переменной через указатель (начиная с объявления указателя).

5. Что такое адресная арифметика?

6. Какие ограничения действуют в адресной арифметике на операцию «присваивания»?

7. Какие ограничения действуют в адресной арифметике на операции «инкремента» и «декремента»?

8. Какие ограничения действуют в адресной арифметике на «сложение» и «вычитание»?
9. Что представляет собой массив данных с точки зрения указателей?
10. Перечислите три направления применения указателей в параметрах функций.
11. Как функция может обратиться к памяти, находящейся за её пределами? Приведите пример.

Тема 13 Язык программирования Object Pascal

Средства текущего контроля для проведения опроса:

1. Язык программирования Object Pascal.
2. Операторы.
3. Процедуры и функции.
4. Строки.
5. Структурированные типы.
6. Тип variant.
7. Динамическая память и указатели.

Практическая работа № 9 Разработка и реализация программ с файлами данных

Цель работы: Научиться составлять алгоритмы структуры типа файл

Вопросы по практической работе 9

1. Файлы. Файловые типы. Понятие прямого и последовательного методов доступа.
2. Файловая система Турбо Паскаля: понятие логического и физического файлов.
3. Текстовые файлы.
4. Типизированные файлы. Процедуры и функции для организации прямого метода доступа.
5. Алгоритмы формирования, просмотра и корректировки текстового и типизированного файлов.
6. Бестиповые файлы.

Тема 14 PascalABC.NET: структура проекта; модули, формы

Средства текущего контроля для проведения опроса:

1. Методы.
2. Возврат из метода.
3. Уровни доступа.
4. Использование параметров.
5. Исключение недоступного кода.
6. Конструкторы.
7. Параметризованные конструкторы.
8. Сборка мусора и деструкторы.
9. Классы потоков.
10. Консольный ввод-вывод.
11. Класс FileStream.
12. Запись в файл.
13. Символьный ввод-вывод в файл.
14. Применение класса StreamWriter.
15. Применение класса StreamReader

Практическая работа № 10 Динамическое программирование

Цель работы: Изучить особенности работы с динамическими переменными, работы с оперативной памятью, познакомиться со стандартными процедурами и функциями для работы с динамическими переменными

Вопросы по практической работе 10

1. Для каких оптимизационных задач применяется метод динамического программирования?
2. В чем заключается суть метода динамического программирования?
3. Сформулируйте принцип оптимальности Беллмана.
4. Что является целевой функцией в задаче о кратчайшем маршруте?
5. Какой параметр определяет состояние системы на каждом шаге?

Тема 15 Понятия объекта, класса

Средства текущего контроля для проведения опроса:

1. Событийное управление.
2. Свойства окна.
3. Элементы интерфейса.
4. События на форме.
5. Ошибки на форме.
6. События кнопки Button.
7. Поле ввода TextBox.
8. События TextBox.
9. Виды классов, индекаторы и свойства.
10. Статические члены класса.
11. Константы и неизменяемые поля.
12. Статические неизменяемые поля.
13. Вложенные классы.
14. Наследование.
15. Инициализаторы конструкторов.
16. Использование интерфейсов.
17. Изолированные классы.
18. Абстрактные классы.
19. Перегрузка.

Практическая работа № 11 Разработка рекурсивных алгоритмов

Цель работы: Изучить особенности работы с рекурсивными процедурами и функциями.

Вопросы по практической работе 11

1. Что такое рекурсивная функция?
2. Что такое прямая рекурсия?
3. Что такое косвенная рекурсия?
4. На каком-то шаге должен быть прекращен дальнейший вызов рекурсивной функции?

Тема 16 Виды классов. ООП в деталях и примерах. Наследование. Разработка Windows-приложений

Средства текущего контроля для проведения опроса:

1. Классы и объекты. Описание класса
2. Конструкторы и создание объекта
3. Методы и свойства класса
4. Одиночное наследование.
5. Отношение наследования.
6. Структурные отношения.
7. Наследование.

8. Доступ к членам класса.
9. Организация защищенного доступа.
10. Конструкторы и наследование.
11. Вызов конструкторов базового класса.
12. Наследование и сокрытие имен.
13. Создание многоуровневой иерархии классов.
14. Порядок вызова конструкторов.
15. Ссылки на базовый класс и объекты производных классов.

Практическая работа № 12 Реализация рекурсивных структур. Применение рекурсивных алгоритмов для различных сортировок

Цель работы: Изучить особенности работы с рекурсивными процедурами и функциями

Вопросы по практической работе 12

1. Можно ли случай косвенной рекурсии свести к прямой рекурсии?
2. Может ли рекурсивная база содержать несколько тривиальных случаев?
3. Являются ли параметры, база и декомпозиция единственными для конкретной задачи?
4. С какой целью в задачах происходит пересмотр или корректировка выбранных параметров, выделенной базы или случая декомпозиции?
5. Является ли рекурсия универсальным способом решения задач?
6. Почему для оценки трудоемкости рекурсивного алгоритма недостаточно одного метода подсчета вершин рекурсивного дерева?

Тема 17 Интерфейсы и структуры.

Средства текущего контроля для проведения опроса:

1. Интерфейсы.
2. Интерфейсные индексы.
3. Интерфейсные свойства.
4. Явные реализации.
5. Что называется интерфейсом?
6. Какие стандартные интерфейсы имеются в библиотеке PascalABC
7. Что из себя представляет структура в PascalABC
8. Чем отличается структура от стандартных классов в PascalABC
9. Для каких целей используются коллекции
10. Структуры

Практическая работа № 13 Модульная структура программы. Отладка модулей

Цель работы: Познакомиться с модульным программированием, со стандартными модулями языка. Научиться программировать модули

Вопросы по практической работе 13

1. Что называется программным средством?
2. Что понимается под надежностью программного средства?
3. Могут ли в надежном программном средстве быть ошибки?
4. Что понимается под технологией программирования?
5. Что такое система? Какие бывают системы?
6. В чем заключается природа ошибок человека при преобразовании информации из одной формы в другую и каковы пути борьбы с этими ошибками?
7. Что входит в разработку программного средства?
8. Что понимают под качеством программного средства?
9. Перечислите критерии качества программного средства.

10. Что понимается под функциональностью (надежностью, легкостью, эффективностью, сопровождаемостью, мобильностью) программного средства?
11. Назовите общие принципы обеспечения надежности программного средства.
12. Какие действия предпринимают для предупреждения ошибок при разработке программных средств?
13. Какие методы используются для борьбы со сложностью программных средств?
14. Как обеспечивается точность перевода документов различными разработчиками программного средства?
15. Что нужно для преодоления барьера между пользователем и разработчиком?
16. Что называют функциональной спецификацией, а что спецификацией качества программного средства?
17. Что такое примитив качества программного средства?
18. Перечислите примитивы качества программного средства. С какими критериями качества они связаны?
19. Дайте определения используемым примитивам качества программных средств.
20. Из чего состоит функциональная спецификация?
21. Что такое модульное программирование (программный модуль)?
22. Какие критерии используются для оценки программного модуля?
23. Чем измеряется размер программного модуля?
24. Что такое прочность программного модуля? Приведите примеры различных по степени прочности модулей.
25. Что такое сцепление модуля?
26. Какой вид сцепления программных модулей рекомендуется современной технологией программирования?
27. Какой модуль называется рутинным (зависящим от предыстории)?
28. В чем состоит метод восходящей (нисходящей) разработки?
29. Какие методы используются для контроля структуры программы? В чем их суть?
30. Назовите порядок разработки программного модуля. Что делается на каждом шаге разработки?
31. Какое программирование называют структурным?
32. Для чего используется язык псевдокода?
33. Какое программирование называется защитным? В чем оно заключается?
34. Перечислите рекомендации по обеспечению ясности текста (эффективности) модуля.
35. Какие методы применяют для контроля программного модуля?
36. Что понимается под отладкой (тестированием) программного средства?
37. Какие виды отладки программных средств приняты в нашей стране?
38. Какие рекомендации по отладке программного средства Вы знаете?
39. Как осуществляется автономное тестирование (тестирование при комплексной отладке)?

Тема 18 Связанные списки на основе рекурсивных данных

Средства текущего контроля для проведения опроса:

1. Основные понятия о списках.
2. Простые списки.
3. Изменение размеров массивов.
4. Список переменного размера
5. Класс SimpleList.
6. Неупорядоченные списки.
7. Связанные списки.

8. Добавление элементов.
9. Удаление элементов.
10. Стеки.
11. Стеки на связанных списках.
12. Очереди
13. Циклические очереди.
14. Очереди на основе связанных списков.
15. Очереди с приоритетом
16. Многопоточные очереди.

Практическая работа № 14 Разработка и реализация модульных программ

Цель работы: Познакомиться с модульным программированием, со стандартными модулями Pascal ABS

Вопросы по практической работе 14

1. Делегаты.
2. Групповая адресация.
3. Групповое преобразование делегируемых методов.
4. Ковариантность и контравариантность.
5. Анонимные методы.
6. Передача аргументов анонимному методу.
7. Возврат значения из анонимного метода.
8. Лямбда-выражения.
9. Одиночные лямбда-выражения.
10. Блочные лямбда-выражения.

Тестовые задания

1. Какие программы можно отнести к системному программному обеспечению:
 1. операционные системы;
 2. прикладные программы;
 3. игровые программы.
2. Какие программы можно отнести к системному ПО:
 1. драйверы;
 2. текстовые редакторы;
 3. электронные таблицы;
 4. графические редакторы.
3. Специфические особенности ПО как продукта:
 1. продажа по ценам ниже себестоимости (лицензирование);
 2. низкие материальные затраты при создании программ;
 3. возможность создание программ небольшие коллективом или даже одним человеком;
 4. разнообразие решаемых задач с помощью программных средств.
4. Какие программы можно отнести к системному ПО:
 1. программа расчета заработной платы;
 2. электронные таблицы;
 3. СУБД (системы управления базами данных).
5. Какие программы нельзя отнести к системному ПО:
 1. игровые программы;
 2. компиляторы языков программирования;
 3. операционные системы;
 4. системы управления базами данных.
6. Какие программы можно отнести к прикладному программному обеспечению:

1. электронные таблицы;
 2. таблицы решений;
 3. СУБД (системы управления базами данных).
7. Какие программы можно отнести к прикладному ПО:
1. программа расчета заработной платы;
 2. диспетчер программ;
 3. программа «Проводник» (Explorer).
8. Какие программы нельзя отнести к прикладному ПО:
1. компиляторы и (или) интерпретаторы;
 2. текстовые и (или) графические редакторы;
 3. электронные таблицы.
9. Можно ли отнести операционную систему к программному обеспечению:
1. да;
 2. нет.
10. Можно ли отнести операционную систему к прикладному программному обеспечению:
1. да;
 2. нет.
11. Специфические особенности ПО как продукта:
1. низкие затраты при дублировании;
 2. универсальность;
 3. простота эксплуатации;
 4. наличие поддержки (сопровождения) со стороны разработчика.
12. Какие программы можно отнести к системному ПО:
1. утилиты;
 2. экономические программы;
 3. статистические программы;
 4. мультимедийные программы.
13. Этап, занимающий наибольшее время, в жизненном цикле программы:
1. сопровождение;
 2. проектирование;
 3. тестирование;
 4. программирование;
 5. формулировка требований.
14. Этап, занимающий наибольшее время, при разработке программы:
1. тестирование;
 2. сопровождение;
 3. проектирование;
 4. программирование;
 5. формулировка требований.
15. Первый этап в жизненном цикле программы:
1. формулирование требований;
 2. анализ требований;
 3. проектирование;
 4. автономное тестирование;
 5. комплексное тестирование.
16. Один из необязательных этапов жизненного цикла программы:
1. оптимизация;
 2. проектирование;
 3. тестирование;
 4. программирование;

5. анализ требований.
17. Самый большой этап в жизненном цикле программы:
 1. эксплуатация;
 2. изучение предметной области;
 3. программирование;
 4. тестирование;
 5. корректировка ошибок.
18. Какой этап выполняется раньше:
 1. отладка;
 2. тестирование.
19. Какой этап выполняется раньше:
 1. отладка;
 2. оптимизация;
 3. программирование;
 4. тестирование.
20. Что выполняется раньше:
 1. компиляция;
 2. отладка;
 3. компоновка;
 4. тестирование.
21. Что выполняется раньше:
 1. проектирование;
 2. программирование;
 3. отладка;
 4. тестирование.
22. В стадии разработки программы не входит:
 1. автоматизация программирования;
 2. постановка задачи;
 3. составление спецификаций;
 4. эскизный проект;
 5. тестирование.
23. Самый важный критерий качества программы:
 1. работоспособность;
 2. надежность;
 3. эффективность;
 4. быстрое действие;
 5. простота эксплуатации.
24. Способы оценки качества:
 1. сравнение с аналогами;
 2. наличие документации;
 3. оптимизация программы;
 4. структурирование алгоритма.
25. Существует ли связь между эффективностью и оптимизацией программы:
 1. да;
 2. нет.
26. Наиболее важный критерий качества:
 1. надежность;
 2. быстрое действие;
 3. удобство в эксплуатации;
 4. удобный интерфейс;
 5. эффективность.

27. Способы оценки надежности:
1. тестирование;
 2. сравнение с аналогами;
 3. трассировка;
 4. оптимизация.
28. Повышает ли качество программ оптимизация:
1. да;
 2. нет.
29. Существует ли связь между надежностью и быстродействием:
1. нет;
 2. да.
30. В каких единицах можно измерить надежность:
1. отказов/час;
 2. км/час;
 3. Кбайт/сек;
 4. операций/сек.
31. В каких единицах можно измерить быстродействие:
1. отказов/час;
 2. км/час;
 3. Кбайт/сек;
 4. операций/сек.
32. Что относится к этапу программирования:
- А) написание кода программы;
- В) разработка интерфейса;
- С) работоспособность;
- Д) анализ требований.
33. Последовательность этапов программирования:
- А) компилирование, компоновка, отладка;
- В) компоновка, отладка, компилирование;
- С) отладка, компилирование, компоновка;
- Д) компилирование, отладка, компоновка.
- 34) Инструментальные средства программирования:
1. компиляторы, интерпретаторы;
 2. СУБД (системы управления базами данных);
 3. BIOS (базовая система ввода-вывода);
 4. ОС (операционные системы).
35. На языке программирования составляется:
1. исходный код;
 2. исполняемый код;
 3. объектный код;
 4. алгоритм.
36. Правила, которым должна следовать программа это:
1. алгоритм;
 2. структура;
 3. спецификация;
 4. состав информации.
37. Можно ли внутри цикла поместить еще один цикл:
1. да;
 2. нет.
38. Можно ли внутри условного оператора поместить еще одно условие:
1. да;
 2. нет.

39. Можно ли одно большое (длинное) выражение разбить на несколько выражений:
1. да;
 2. нет.
40. Если имеется стандартная функция, нужно ли писать собственную:
1. нет;
 2. да.
41. Доступ, при котором записи файла читаются в физической последовательности, называется:
1. последовательным;
 2. прямым;
 3. простым;
 4. основным.
42. Доступ, при котором записи файла обрабатываются в произвольной последовательности, называется:
1. прямым;
 2. последовательным;
 3. простым;
 4. основным.
43. Методы программирования (укажите НЕ верный ответ):
1. логическое;
 2. структурное;
 3. модульное.
44. Что выполняется раньше:
1. разработка алгоритма;
 2. выбор языка программирования;
 3. написание исходного кода;
 4. компиляция.
45. Можно ли переменным присваивать произвольные идентификаторы:
1. да;
 2. нет.
46. Найдите НЕ правильное условие для создания имен:
1. имена могут содержать пробелы;
 2. длинное имя можно сократить;
 3. из имени лучше выбрасывать гласные;
 4. можно использовать большие буквы.
47. Какие символы не допускаются в именах переменных:
1. пробелы;
 2. цифры;
 3. подчеркивание
48. Можно ли использовать имена, которые уже были использованы в другой программе (модуле):
1. да;
 2. нет.
49. Можно ли ставить знак подчеркивания в начале имени:
1. да, но не рекомендуется;
 2. да, без ограничений;
 3. нет.
50. Как называется способ составления имен переменных, когда в начале имени сообщается тип переменной:
5. прямым указанием;
 6. венгерской нотацией;

7. структурным программированием;
8. поляризацией.
51. Можно ли писать комментарии в отдельной строке:
 1. да;
 2. нет.
52. Транслируются ли комментарии:
 1. да;
 2. нет.
53. Наличие комментариев позволяет:
 1. быстрее найти ошибки в программе;
 2. быстрее писать программы;
 3. быстрее выполнять программы.
54. Наличие комментариев позволяет:
 1. легче разобраться в программе;
 2. применять сложные структуры;
 3. увеличить быстродействие.
55. Наличие комментариев позволяет:
 1. улучшить читабельность программы;
 2. улучшить эксплуатацию программы;
 3. повысить надежность программы.
56. Что определяет выбор языка программирования:
 1. область приложения;
 2. знание языка;
 3. наличие дополнительных библиотек.
57. Возможно ли комбинирование языков программирования в рамках одной задачи:
 1. да;
 2. нет.
58. Допустимо ли комбинирование языков программирования в рамках одной задачи:
 1. да;
 2. нет.
59. Для каких задач характерно использование большого количества исходных данных, выполнение операций поиска, группировки:
 1. для экономических задач;
 2. для системных задач;
 3. для инженерных задач.
60. Для каких задач характерен большой объем вычислений, использование сложного математического аппарата:
 1. для инженерных задач;
 2. для системных задач;
 3. для экономических задач.
61. На каком этапе производится выбор языка программирования:
 1. проектирование;
 2. программирование;
 3. отладка;
 4. тестирование.
62. Можно ли использовать комбинацию языков программирования в рамках одного проекта:
 1. да;
 2. нет.
63. Для решения экономических задач характерно применение:

1. СУБД (систем управления базами данных);
 2. языков высокого уровня;
 3. языков низкого уровня;
 4. применение сложных математических расчетов.
64. Для решения инженерных задач характерно применение:
1. САПР (систем автоматизированного проектирования);
 2. СУБД (систем управления базами данных);
 3. ОС (операционных систем).
65. Причины синтаксических ошибок:
1. плохое знание языка программирования;
 2. ошибки в исходных данных;
 3. ошибки, допущенные на более ранних этапах;
 4. неправильное применение процедуры тестирования.
66. Когда можно обнаружить синтаксические ошибки:
1. при компиляции;
 2. при отладке;
 3. при тестировании;
 4. на этапе проектирования;
 5. при эксплуатации.
67. Ошибки компоновки заключаются в том, что:
1. указано внешнее имя, но не объявлено;
 2. неправильно использовано зарезервированное слово;
 3. составлено неверное выражение;
 4. указан неверный тип переменной.
68. Могут ли проявиться ошибки при изменении условий эксплуатации:
1. да;
 2. нет.
69. Могут ли проявиться ошибки при изменении в предметной области:
1. да;
 2. нет.
70. Возможно ли программирование с защитой от ошибок:
1. да;
 2. нет.
71. Есть ли недостатки программирования с защитой от ошибок:
1. да;
 2. нет.
72. Защитное программирование это:
1. встраивание в программу отладочных средств;
 2. создание задач защищенных от копирования;
 3. разделение доступа в программе;
 4. использование паролей;
 5. оформление авторских прав на программу.
73. Вид ошибки с неправильным написанием служебных слов (операторов):
1. синтаксическая;
 2. семантическая;
 3. логическая;
 4. символьная.
74. Вид ошибки с неправильным использованием служебных слов (операторов):
1. семантическая;
 2. синтаксическая;
 3. логическая;
 4. символьная.

75. Ошибки при написании программы бывают:

1. синтаксические;
2. орфографические;
3. лексические;
4. фонетические;
5. морфологические.

76. Процедура поиска ошибки, когда известно, что она есть это:

1. отладка;
2. тестирование;
3. компоновка;
4. транзакция;
5. трансляция.

77. Программа для просмотра значений переменных при выполнении программы:

7. отладчик;
8. компилятор;
9. интерпретатор;
10. трассировка;
11. тестирование.

78. Отладка – это:

1. процедура поиска ошибок, когда известно, что ошибка есть;
2. определение списка параметров;
3. правило вызова процедур (функций);
4. составление блок-схемы алгоритма.

79. Когда программист может проследить последовательность выполнения команд программы:

1. при трассировке;
2. при тестировании;
3. при компиляции;
4. при выполнении программы;
5. при компоновке.

80. На каком этапе создания программы могут появиться синтаксические ошибки:

1. программирование;
2. проектирование;
3. анализ требований;
4. тестирование.

81. Когда приступают к тестированию программы:

1. когда программа уже закончена;
2. после постановки задачи;
3. на этапе программирования;
4. на этапе проектирования;
5. после составления спецификаций,

82. Тестирование бывает:

1. автономное;
2. инструментальное;
3. визуальное;
4. алгоритмическое.

83. Тестирование бывает:

1. комплексное;
2. инструментальное;
3. визуальное;
4. алгоритмическое.

84. Существует ли различие между отладкой и тестированием:
1. да;
 2. нет.
85. При комплексном тестировании проверяются:
1. согласованность работы отдельных частей программы;
 2. правильность работы отдельных частей программы;
 3. быстродействие программы;
 4. эффективность программы.
86. Чему нужно уделять больше времени, чтобы получить хорошую программу:
1. тестированию;
 2. программированию;
 3. отладке;
 4. проектированию.
87. Процесс исполнения программы с целью обнаружения ошибок:
1. тестирование;
 2. кодирование;
 3. сопровождение;
 4. проектирование.
88. Автономное тестирование это:
1. тестирование отдельных частей программы;
 2. инструментальное средство отладки;
 3. составление блок-схем;
 4. пошаговая проверка выполнения программы.
89. Трассировка это:
1. проверка пошагового выполнения программы;
 2. тестирование исходного кода;
 3. отладка модуля;
 4. составление блок-схемы алгоритма.
90. Локализация ошибки:
1. определение места возникновения ошибки;
 2. определение причин ошибки;
 3. обнаружение причин ошибки;
 4. исправление ошибки.
91. Назначение тестирования:
1. повышение надежности программы;
 2. обнаружение ошибок;
 3. повышение эффективности программы;
 4. улучшение эксплуатационных характеристик;
 5. приведение программы к структурированному виду.
92. Назначение отладки:
1. поиск причин существующих ошибок;
 2. поиск возможных ошибок;
 3. составление спецификаций;
 4. разработка алгоритма.
93. Инструментальные средства отладки (НЕ правильный ответ):
1. компиляторы;
 2. отладчики;
 3. трассировка.
94. Отладка программ это:
1. локализация и исправление ошибок;
 2. алгоритмизация программирования;
 3. компиляция и компоновка.

95. Что выполняется раньше, автономная или комплексная отладка:
1. автономная;
 2. комплексная.
96. Что выполняется раньше, отладка или тестирование:
1. отладка;
 2. тестирование.
97. Что такое автоматизация программирования:
1. создание исходного кода программными средствами;
 2. создание исходного кода при помощи компилятора;
 3. создание исходного кода без разработки алгоритма.
98. В чем сущность автоматизации программирования:
1. создание программы без написания ее текста;
 2. получение готовой программы без выполнения компоновки;
 3. в отсутствие компиляции.
99. Возможна ли автоматизация программирования:
1. да;
 2. нет.
100. Создание исполняемого кода программы без написания исходного кода называется:
1. составлением спецификаций;
 2. отладкой;
 3. проектированием.
 4. автоматизацией программирования;
101. Одно из преимуществ автоматизации программирования:
1. наглядное программирование с визуальным контролем;
 2. получение стандартной программы;
 3. создание программы с оптимальным кодом.
102. Один из методов автоматизации программирования:
1. структурное программирование;
 2. модульное программирование;
 3. визуальное программирование;
 4. объектно-ориентированное программирование.
103. Влияет ли автоматизация программирования на эффективность программы:
1. нет;
 2. да
104. Автоматизация программирования позволяет:
1. повысить надежность программы;
 2. сократить время разработки программы;
 3. повысить быстродействие программы.
105. Позволяет ли автоматизация программирования всегда создавать эффективные программы:
1. да.
 2. нет;
106. Позволяет ли автоматизация программирования всегда создавать надежные программы:
- А) нет; В) да.
107. Недостаток автоматизации программирования; В) низкое быстродействие; А) большой размер программы; С) сложность программы.
108. Возможны ли ошибки при автоматизации программирования: А) да; В) нет.
109. Нужно ли выполнять тестирование при автоматизации программирования: А) да;

- В) нет.
110. Выполняется ли процедура компиляции при автоматизации программирования: А) да;
В) нет.
111. Что легко поддается автоматизации: А) интерфейс;
В) работа с файлами;
С) сложные логические задачи; D) алгоритмизация.
112. Относится ли визуальное программирование к средствам автоматизации: А) да;
В) нет.
113. Нахождение наилучшего варианта из множества возможных: А) оптимизация;
В) тестирование; С) автоматизация; D) отладка;
Е) сопровождение.
114. Что такое оптимизация программ:
А) улучшение работы существующей программы; В) создание удобного интерфейса пользователя;
С) разработка модульной конструкции программы;
D) применение методов объектно-ориентированного программирования.
115. Критерии оптимизации:
А) время выполнения или размер требуемой памяти; В) размер программы и ее эффективность;
С) независимость модулей;
D) качество программы, ее надежность.
116. Критерии оптимизации:
А) эффективность использования ресурсов; В) структурирование алгоритма;
С) структурирование программы.
117. Возможна ли оптимизация программ без участия программиста: А) да;
В) нет.
118. Возможна ли оптимизация циклов: А) да;
В) нет.
119. В чем заключается оптимизация условных выражений: А) в изменении порядка следования элементов выражения; В) в использовании простых логических выражений;
С) в использовании сложных логических выражений; D) в использовании операций AND, OR и NOT.
120. Оптимизация циклов заключается в:
А) уменьшении количества повторений тела цикла; В) просмотре задачи с другой стороны;
С) упрощении задачи за счет включения логических операций.
121. Оптимизация программы это: А) модификация;
В) отладка;
С) повышение сложности программы; D) уменьшение сложности программы.
122. Критерии оптимизации программы:
А) быстродействие или размер программы; В) быстродействие и размер программы;
С) надежность или эффективность; D) надежность и эффективность.
123. Результат оптимизации программы:
А) эффективность; В) надежность;
С) машино-независимость; D) мобильность.
124. Сущность оптимизации циклов:
А) сокращение количества повторений выполнения тела цикла; В) сокращение тела цикла;
С) представление циклов в виде блок-схем; D) трассировка циклов;

- Е) поиск ошибок в циклах.
125. В чем сущность модульного программирования:
А) в разбиении программы на отдельные функционально независимые части; В) в разбиении программы на отдельные равные части;
С) в разбиение программы на процедуры и функции;
126. Можно ли сочетать модульное и структурное программирование: А) да;
В) нет.
127. Может ли модуль включать несколько процедур или функций: А) да;
В) нет.
128. Рекомендуемые размеры модулей: А) небольшие;
В) большие; С) равные;
D) фиксированной длины.
129. В чем заключается независимость модуля:
А) в написании, отладке и тестировании независимо от остальных модулей; В) в разработке и написании независимо от других модулей;
С) в независимости от работы основной программы.
130. При модульном программировании желательно, чтобы модуль имел: А) большой размер;
В) небольшой размер;
С) фиксированный размер; D) любой размер.
131. Модульное программирование это:
А) разбиение программы на отдельные части; В) структурирование;
С) использование стандартных процедур и функций.
132. Можно ли использовать оператор GO TO в модульных программах: А) можно;
В) нельзя.
133. Разрешается ли использование циклов при модульном программировании: А) да;
В) нет.
134. Разрешается ли использование условных операторов при модульном программировании:
А) да; В) нет.
135. Сократится ли размер программы, если ее написать в виде набора модулей: А) нет;
В) да.
136. Достоинство модульного программирования:
А) создание программы по частям в произвольном порядке; В) не требует компоновки;
С) всегда дает эффективные программы; D) снижает количество ошибок.
137. Недостаток модульного программирования: А) увеличивает трудоемкость программирования;
В) усложняет процедуру комплексного тестирования; С) снижает быстродействие программы;
D) не позволяет выполнять оптимизацию программы.
138. Достоинство модульного программирования:
А) возможность приступить к тестированию до завершения написания всей программы; В) не требует комплексного тестирования;
С) уменьшает размер программы;
D) повышает надежность программы.
139. Допустимо ли использование оператора GO TO при структурном программировании: А) нет;
В) да.
140. Можно ли сочетать структурное программирование с модульным: А) можно;

- В) нельзя;
 - С) только в особых случаях.
141. Любую ли программу можно привести к структурированному виду: А) любую; В) не все; С) нельзя.
142. Можно ли использовать оператор GO TO в структурированных программах:
- А) можно;
 - В) нельзя;
 - С) только в особых случаях.
143. Возможно, ли преобразовать неструктурированную программу к структурному виду: А) да; В) нет.
144. Возможно ли программирование без оператора GO TO: А) да; В) нет.
145. При структурном программировании задача выполняется: А) поэтапным разбиением на более легкие задачи; В) без участия программиста; С) объединением отдельных модулей программы.
146. Разрешается ли использование оператора GO TO при структурном программировании: А) нет; В) да; С) иногда.
147. Разрешается ли использование циклов при структурном программировании: А) да; В) нет.
148. Разрешается ли использование оператора IF при структурном программировании: А) да; В) нет.
149. Программирование без GO TO применяется. при: А) структурном программировании; В) модульном программировании; С) объектно-ориентированном программировании; D) все ответы верные.
150. Достоинство структурного программирования: А) можно приступить к комплексному тестированию на раннем этапе разработки; В) можно приступить к автономному тестированию на раннем этапе разработки; С) нет необходимости выполнять тестирование; D) можно пренебречь отладкой.

10.2 Критерии оценки результатов текущего контроля освоения дисциплины

Критерии оценки устного опроса

Оценка, уровень достижения компетенций	Описание критериев
Отлично, высокий	Обучающийся демонстрирует уверенное знание материала, полно излагает материал, дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; излагает материал последовательно и правильно с точки зрения норм литературного языка
Хорошо, продвинутый	Обучающийся демонстрирует уверенное знание материала, но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого.
Удовлетворительно, пороговый	Обучающийся обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.
Неудовлетворительно, компетенция не освоена	Обучающийся демонстрирует незнание большей части соответствующего вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

Критерии оценки практической работы

Оценка «отлично» – ставится, если обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, определяет взаимосвязи между показателями задачи, даёт правильный алгоритм решения, определяет междисциплинарные связи по условию задания. А также, если обучающийся имеет глубокие знания учебного материала по теме практической работы, показывает усвоение взаимосвязи основных понятий используемых в работе, смог ответить на все уточняющие и дополнительные вопросы.

Оценка «хорошо» – ставится, если обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, допуская незначительные неточности при решении задач, имея неполное понимание междисциплинарных связей при правильном выборе алгоритма решения задания. А также, если обучающийся показал знание учебного материала, усвоил основную литературу, смог ответить почти полно на все заданные дополнительные и уточняющие вопросы.

Оценка «удовлетворительно» – ставится, если обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, требующий наводящих вопросов преподавателя, выбор алгоритма решения задачи возможен при наводящих вопросах преподавателя. А также, если обучающийся в целом освоил материал практической работы, ответил не на все уточняющие и дополнительные вопросы.

Оценка «неудовлетворительно» – ставится, если обучающийся дает неверную оценку ситуации, неправильно выбирает алгоритм действий. А также, если он имеет существенные пробелы в знаниях основного учебного материала практической работы, который полностью не раскрыл содержание вопросов, не смог ответить на уточняющие и дополнительные вопросы.

Критерии оценки тестовых заданий

Оценка, уровень достижения компетенций	Описание критериев
Отлично, высокий	Содержание правильных ответов в тесте не менее 90%
Хорошо, продвинутый	Содержание правильных ответов в тесте не менее 75%
Удовлетворительно, пороговый	Содержание правильных ответов в тесте не менее 50%
Неудовлетворительно, компетенция не освоена	Содержание правильных ответов в тесте менее 50%

10.3. Оценочные материалы для промежуточной аттестации по дисциплине

Вопросы для подготовки к экзамену

1. Основные конструкции алгоритмического языка
2. Жизненный цикл программы
3. Особенности языка программирования
4. Структура программы в языке
5. Понятие переменной
6. Объявление переменных.
7. Типы данных, определяемые пользователем.
8. Инициализация переменных.
9. Основные типы переменных
10. Синтаксис объявления имени переменной
11. Понятие константы. Объявление.
12. Понятие константы. Инициализация.
13. Простейшие арифметические операции
14. Операция %. Ее особенности.
15. Основные математические функции
16. Принципы структурного программирования
17. Оператор множественного ветвления switch
18. Условный оператор if-else
19. Вложенный оператор if.
20. Классификация циклов
21. Операции инкремента и декремента.
22. Оператор while()
23. Оператор For.
24. Оператор do...while()
25. Принципы модульного программирования
26. Описание функции
27. Вызов функции.
28. Прототипы функций.
29. Передача параметров по имени.
30. Передача параметров по адресу.
31. Данные типа «указатель».
32. Операции для работы с указателями
33. Объявление и инициализация одномерного массива.
34. Объявление и инициализация двумерного массива.

35. Доступ к элементам одномерного массива по индексу.
36. Доступ к элементам одномерного массива по адресу.
37. Доступ к элементам двумерного массива по индексу.
38. Доступ к элементам двумерного массива по адресу.
39. Принципы объектно-ориентированного программирования. Инкапсуляция.
40. Принципы объектно-ориентированного программирования. Полиморфизм.
41. Принципы объектно-ориентированного программирования. Наследование.
42. Структуры. Объявление.
43. Структуры. Инициализация.
44. Прямой доступ к элементам структуры.
45. Косвенный доступ к элементам структуры.
46. Массивы структур. Объявление.
47. Массивы структур. Инициализация.
48. Классы. Объявление открытых переменных.
49. Классы. Инициализация открытых переменных.
50. Классы. Объявление закрытых переменных.
51. Классы. Инициализация закрытых переменных.
52. Спецификаторы доступа.
53. Функции-члены класса.
54. Связывание функций.
55. Встроенные функции.
56. Конструкторы . Основные понятия.
57. Конструкторы по умолчанию.
58. Конструкторы с параметрами.
59. Стандартный класс string. Основные понятия.
60. Основные операции класса string.

Практические задания

1. На плоскости заданы n точек своими координатами $(x_1, y_1), (x_2, y_2)$. Составить программу вычисления максимального внутреннего и минимального внешнего радиусов кольца с центром в начале координат, содержащего все точки. Координаты точек считывать из файла.
2. Составить программу, которая вводит натуральное число N и выдает все трехзначные числа, сумма цифр которых равна $N*3$.
3. Составить программу, которая считывает с командной строки имя файла и выводит на экран количество символов, строк и слов в этом файле.
4. Программа должна считывать с командной строки имена двух файлов и выводить на экран те строки этих файлов, которые отличаются друг от друга.
5. Программа, принимающая имя файла и выводящая его строки, перед каждой из которых записывается ее номер.
6. Программа считывает из файла матрицу и выводит на экран суммы столбцов этой матрицы.
7. Программа считывает с командной строки имя файла и слово и проверяет, встречается ли такое слово в файле.
8. Программа принимает имя файла, сортирует его строки и выдает их на экран.
9. Программа считывает с командной строки имя файла и выводит на экран все слова, встречающиеся в этом файле, без повторений.
10. Программа считывает с командной строки имя файла и выводит на экран наиболее часто встречающееся в нем слово.

11. Составить программу, которая запрашивает у пользователя двузначное целое число, вводит его и отображает на экране величину числа словами. Например, введено - 12. Результат: минус двенадцать.

12. Программа принимает имя файла и символ и выводит на экран то есть строки файла, которые начинаются с указанного символа.

13. Программа для преобразования единиц измерения предназначена для преобразования физических величин друг в друга (метры в километры или сантиметры, килограммы в тонны, граммы и фунты). Программа принимает в командной строке число, исходную единицу измерения и целевую единицу измерения и выводит на экран преобразованное число. Единицы измерения задавать строками ("m" - для метров, "km" - километры, "cm" - сантиметры, "kg" - килограммы, "t" - тонны, "g" - граммы, "p" - фунты)

14. Программа, которая по заданному числу N печатает список всех простых чисел, не превышающих N.

15. Программа, которая по заданному числу N печатает список всех совершенных чисел, не превышающих N. Совершенным называется число, которое равно сумме своих делителей, например $6 = 1+2+3$ или $24 = 1 + 2 + 3 + 4 + 6 + 8$.

16. Программа считывает с командной строки имя файла и два слова и распечатывает на экране содержимое файла, в котором все вхождения первого слова заменены на второе.

17. Программа, которая выводит на экран все алфавитно-цифровые последовательности, содержащиеся в указанном файле. Алфавитно-цифровой называется непрерывная последовательность цифр и букв латинского алфавита. Пример: если файл содержит символы `asdf+^&@qwerty!@#` то результатом работы программы должно быть: `asdfqwerty`

18. Программа принимает в командной строке имя файла и один из двух символов 'u' или 'l'. В зависимости от того, какой из символов передан, она преобразует содержимое файла к верхнему или нижнему регистру и распечатывает на экране.

19. Программа считывает две матрицы из файлов и записывает в третий файл матрицу, являющуюся их суммой.

20. Программа считывает из одного файла матрицу, а из другого – вектор и записывает в третий файл результат умножения матрицы на вектор.

10.4 Показатели, критерии и шкала оценивания ответов на экзамене

Оценка, уровень достижения компетенций	Описание критериев
Отлично, высокий	Обучающийся показал полные и глубокие знания программного материала, логично и аргументировано ответил на все вопросы экзаменационного билета, а также на дополнительные вопросы, способен самостоятельно решать сложные задачи дисциплины, практическое задание выполнено без ошибок
Хорошо, продвинутый	Обучающийся твердо знает программный материал, грамотно его излагает, не допускает существенных неточностей в ответе, достаточно полно ответил на вопросы экзаменационного билета и дополнительные вопросы, способен самостоятельно решать стандартные задачи дисциплины, практическое задание выполнено без ошибок
Удовлетворительно, пороговый	Обучающийся показал знание только основ программного материала, усвоил его поверхностно, но не допускал грубых ошибок или неточностей, требует наводящих вопросов для правильного ответа, не ответил на дополнительные вопросы, спо-

	способен решать стандартные задачи дисциплины с помощью преподавателя, практическое задание в целом выполнено, но содержит незначительные недостатки
Неудовлетворительно, компетенция не освоена	Обучающийся не знает основ программного материала, допускает грубые ошибки в ответе, не способен решать стандартные задачи дисциплины даже с помощью преподавателя, практическое задание выполнено частично или не выполнено