

**Автономная некоммерческая профессиональная образовательная  
организация  
«Калининградский колледж управления»**

Лист актуализации рабочей программы дисциплины<sup>1</sup>

ОП.06 «Основы алгоритмизации и программирования»  
Специальность: 09.02.04 - «Информационные системы (по отраслям)»

В целях актуализации образовательной программы с учетом появления новых учебников, учебных пособий и других учебно-методических материалов в рабочую программу внесены следующие изменения (дополнения):

1. п. 5.2 Лицензионное программное обеспечение - проведена актуализация лицензионного программного обеспечения;
2. п. 6 Оценочные средства и методические материалы по итогам освоения дисциплины внесено дополнение, что при разработке оценочных средств преподавателем используются базы данных педагогических измерительных материалов, предоставленных ООО «Научно-исследовательский институт мониторинга качества образования»;
3. п. 7 обновлен список основной учебной литературы;
4. п. 8. Дополнительные ресурсы информационно - телекоммуникационной сети «Интернет» необходимые для освоения дисциплины - внесен ресурс <https://i-exam.ru/> - Единый портал интернет-тестирования в сфере образования.
5. в Приложение 1 к РПД п. 6.2 (Методические рекомендации и указания) – актуализированы рекомендации по проведению учебных занятий с обучающимися с применением электронного обучения и дистанционных образовательных технологий.

Разработчик: *Воробейкина И.В.*

(ФИО, ученая степень, ученое звание)

«17» мая 2023 г.

(дата)

Изменения (дополнения) в рабочую программу рассмотрены и утверждены на заседании учебно-методического совета, протокол № 57 от «25» мая 2023 г.

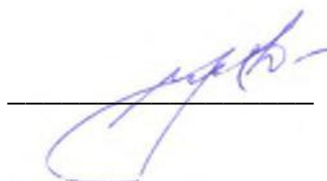
СОГЛАСОВАНО:

Руководитель ОПОП



Шульгина Н.В.

Начальник УМУ



Усенок С.С.

26 мая 2023 г.



<sup>1</sup> Лист актуализации сдается в электронном виде в Учебный отдел АНПОО «ККУ»

**Автономная некоммерческая профессиональная  
образовательная организация  
«КАЛИНИНГРАДСКИЙ КОЛЛЕДЖ УПРАВЛЕНИЯ»**

Утверждено  
Учебно-методическим советом Колледжа  
протокол заседания  
№ 24 от 20 февраля 2020 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ  
ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ  
(ОП.06)**

По специальности	<b>09.02.04 Информационные системы (по отраслям)</b>
Квалификация	<b>Техник по информационным системам</b>
Форма обучения	<b>Очная</b>

Рабочий учебный план по специальности  
утвержден директором 05 ноября 2019 г.

Калининград

Рабочая программа дисциплины «Основы алгоритмизации и программирования» разработана в соответствии с федеральным государственным образовательным стандартом среднего профессионального образования по специальности 09.02.04 Информационные системы (по отраслям), утвержденным приказом Министерства образования и науки Российской Федерации от 14 мая 2015 г. № 525.

Составитель (автор) - старший преподаватель Воробейкина И.В.

Рецензент - старший преподаватель Подтопельный В.В.

Рабочая программа дисциплины рассмотрена и одобрена на заседании Учебно-методического совета колледжа, протокол № 24 от 20 февраля 2020 г.

Регистрационный номер ИС 34/20

<b>Содержание</b>		<b>Стр.</b>
1	Цели и задачи освоения дисциплины	4
2	Место дисциплины в структуре ППСЗ	4
3	Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы	4
4	Объем, структура и содержание дисциплины с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся.	5
4.1.	Объем дисциплины	6
4.2.	Структура дисциплины	6
4.2.1.	Теоретические занятия - занятия лекционного типа	6
4.2.2.	Занятия семинарского типа	7
4.2.3.	Самостоятельная работа	8
5	Перечень образовательных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, современных профессиональных баз данных и информационных справочных систем	9
5.1.	Образовательные технологии	9
5.2.	Лицензионное программное обеспечение	9
5.3.	Современные профессиональные базы данных	9
5.4.	Информационные справочные системы	10
6	Фонд оценочных средств и методические материалы по освоению дисциплины	10
7	Основная и дополнительная учебной литература и электронные образовательные ресурсы, необходимые для освоения дисциплины	10
7.1.	Основная учебная литература	10
7.2.	Дополнительная учебная литература	10
7.3.	Электронные образовательные ресурсы	10
8	Дополнительные ресурсы информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	10
9	Требования к минимальному материально-техническому обеспечению, необходимому для осуществления образовательного процесса по дисциплине	11
	Приложение 1. Фонд оценочных средств	12
	Приложение 2. Методические рекомендации и указания	23

## **1. Цель и задачи освоения дисциплины**

Целями освоения дисциплины «Основы алгоритмизации и программирования» являются формирование у обучающихся представления о языках программирования высокого уровня, а также составлении алгоритмов и их использовании при написании программ.

Задачами курса «ОАиП» являются:

1. знакомство с многообразием современных языков программирования и сред их функционирования;
2. обучение грамотному и осознанному владению системами программирования;
3. изучение номенклатуры средств программирования, используемых в различных предметных областях, и знакомство с их основными функциональными возможностями.

## **2. Место дисциплины в структуре ПСССЗ**

Учебная программа дисциплины «Основы алгоритмизации и программирования» является программой общепрофессиональной дисциплины профессионального цикла программы подготовки специалистов среднего звена 09.02.04 Информационные системы (по отраслям).

Требованием к исходному уровню подготовки обучающихся является уверенное владение материалом следующих учебных дисциплин: Информационные технологии, Операционные системы, Математика. В свою очередь освоение материала дисциплины «Основы алгоритмизации и программирования» обеспечивает преемственность знаний в дальнейшем процессе обучения при освоении следующих дисциплин: Технические средства автоматизации, пм 01 Разработка и эксплуатация автоматизированных информационных систем, а также всех видов практики.

## **3. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы**

В результате изучения обязательной части цикла и освоения дисциплины обучающийся должен уметь:

- использовать языки программирования, строить логически правильные и эффективные программы;

знать:

- общие принципы построения алгоритмов, основные алгоритмические конструкции;
- понятие системы программирования;
- основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти;
- подпрограммы, составление библиотек программ;
- объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов.

Результатами освоения рабочей программы учебной дисциплины является овладение студентами общими компетенциями:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.

ОК 6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.

В процессе изучения дисциплины у обучающихся формируются следующие профессиональные компетенции:

ПК 1.2 Взаимодействовать со специалистами смежного профиля при разработке методов, средств и технологий применения объектов профессиональной деятельности.

ПК 1.3 Производить модификацию отдельных модулей информационной системы в соответствии с рабочим заданием, документировать произведенные изменения.

ПК 2.2. Программировать в соответствии с требованиями технического задания.

ПК 2.3. Применять методики тестирования разрабатываемых приложений.

ПК 2.4. Формировать отчетную документацию по результатам работ.

#### 4. Объем, структура и содержание дисциплины с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам занятий) и на самостоятельную работу обучающихся.

##### 4.1 Объем дисциплины

Объем дисциплины	Всего акад. часов
	для очной формы обучения
Всего академических часов учебных занятий	216
В том числе:	
контактная работа обучающихся с преподавателем (по видам учебных занятий):	144
Лекции	48/
Семинары, практические занятия	96
Практикумы	-
Лабораторные работы	-
Самостоятельная работа	62
Консультация	6
Промежуточная аттестация обучающегося - экзамен	4

##### 4.2. Структура дисциплины

№ п/п	Раздел дисциплины	Семестр	Виды учебной работы, включая самостоятельную работу обучающихся и трудоемкость (в часах ауд/астр)			Формы текущего контроля успеваемости Форма промежуточной аттестации
			Лекции	Практ. зан.	СРС	

	1	2	3	4	5	6
<b>Раздел 1. Основные средства ОАиП</b>						
1.1	Принципы построения алгоритмов.	6	2	4		Входной контроль (контрольная работа)
1.2	Изучение основ языка программирования.	6	2	4	2	План-конспект
1.3	Ввод-вывод, условные операторы if и case.	6	2	6	4	
1.4	Работа со строковым типом данных.	6	4	6	4	
1.5	Циклы for, while, repeat.	6	4	8	4	
<b>Раздел 2. Средства для оптимизации программирования</b>						
2.1	Работа со структурированными типами данных.	6	4	6	4	Текущий контроль (тест)
2.2	Подпрограммы-процедуры.	6	2	6	4	
2.3	Подпрограммы-функции.	6	2	6	4	
2.4	Библиотечные модули.	6	4	6	4	
2.5	Текстовые и типизированные файлы.	6	4	6	4	
2.6	Алгоритмы записи в файл и чтения из файла.	6	2	6	4	
<b>Раздел 3. Работа с динамической памятью</b>						
3.1	Указатели.	6	4	4	4	Рубежный контроль
3.2	Работа с динамической памятью.	6	4	6	6	
3.3	Динамические структуры.	6	4	12	8	
3.4	Основы объектно-ориентированного программирования.	6	4	10	6	
	<b>ИТОГО</b>	6	48	96	62	
Консультация - 6 часов						
Промежуточная аттестация – экзамен (4 часа)						

#### 4.2.1. Теоретические занятия – занятия лекционного типа

№ п/п	Наименование раздела (модуля) дисциплины, темы	Содержание	Кол-во часов
<b>Раздел 1. Основные средства ОАиП</b>			
1	<b>Тема 1.1.</b> Принципы построения алгоритмов.	Алгоритм и его свойства. Основные требования ГОСТа 19.003-80. Графическое изображение алгоритмов линейной, разветвляющейся и циклической структур.	2
2	<b>Тема 1.2.</b> Изучение основ языка программирования.	Характеристика языка Турбо Паскаль. Алфавит языка Паскаль. Структура программы на Паскале. Простые типы данных. Краткая характеристика. Операции и выражения. Оператор присваивания. Составной оператор.	2
3	<b>Тема 1.3.</b> Ввод-вывод, условные операторы if и case.	Процедуры ввода READLN и вывода WRITELN. Условный оператор if. Оператор выбора case. Метки и операторы перехода.	2
4	<b>Тема 1.4.</b> Работа со строковым типом данных.	Строки. Операции над строками. Стандартные процедуры и функции для строк.	4
5	<b>Тема 1.5.</b> Циклы for, while, repeat.	Операторы цикла (с предусловием, с постусловием и с параметрами).	4
<b>Раздел 2. Средства для оптимизации программирования</b>			
6	<b>Тема 2.1.</b> Работа со структурированными типами данных.	Структурированные типы данных. Краткая характеристика. Массивы. Записи. Оператор присоединения With ... do. Множества. Конструкторы множеств. Операции над множествами.	4

7	<b>Тема 2.2.</b> Подпрограммы-процедуры.	Подпрограммы-процедуры. Обращение к процедуре. Параметры-значения. Параметры-переменные.	2
8	<b>Тема 2.3.</b> Подпрограммы-функции.	Подпрограммы-функции. Обращение к функции. Формальные и фактические параметры. Глобальные и локальные параметры.	2
9	<b>Тема 2.4.</b> Библиотечные модули.	Характеристика стандартных модулей Турбо Паскаля. Структура библиотечного модуля. Интерфейсная, исполняемая и иницилирующая части модуля. Компиляция модулей.	4
10	<b>Тема 2.5.</b> Текстовые и типизированные файлы.	Процедуры работы с текстовыми и типизированными файлами. Ошибки открытия файла.	4
11	<b>Тема 2.6.</b> Алгоритмы записи в файл и чтения из файла.	Вывод данных в файл. Чтение данных из файла. Алгоритмы чтения и записи информации.	2
<b>Раздел 3. Работа с динамической памятью</b>			
12	<b>Тема 3.1.</b> Указатели.	Указатели. Операции с указателями.	4
13	<b>Тема 3.2.</b> Работа с динамической памятью.	Обращение к динамической памяти. Создание, использование и удаление динамических переменных.	4
14	<b>Тема 3.3.</b> Динамические структуры.	Стеки, очереди, линейные списки.	4
15	<b>Тема 3.4.</b> Основы объектно-ориентированного программирования.	Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Описание объектов. Экземпляры объектов. Иерархии объектов. Работа с объектами динамической памяти.	4
ИТОГО			48

#### 4.2.2. Занятия семинарского типа

№ п/п	Наименование темы	Содержание учебных вопросов	Кол-во часов
<b>Раздел 1. Основные средства ОаиП</b>			
1	<b>Тема 1.1.</b> Принципы построения алгоритмов.	ПЗ 1.1.1. Изучение графической реализации алгоритма (линейного, разветвляющего, циклического).	4
2	<b>Тема 1.2.</b> Изучение основ языка программирования.	ПЗ 1.2.1. Знакомство со структурой программы на Паскале. Написание простейших программ.	2
		ПЗ 1.2.2. Написание программ с использованием блока var для знакомства с простыми типами данных.	2
3	<b>Тема 1.3.</b> Ввод-вывод, условные операторы if и case.	ПЗ 1.3.1. Написание программ с использованием операторов ввода./вывода write(ln), read(ln).	2
		ПЗ 1.3.2. Написание разветвляющих программ (if, case, goto).	4
4	<b>Тема 1.4.</b> Работа со строковым типом данных.	ПЗ 1.4.1. Написание программ, реализующих работу со строковым типом данных (delete, insert, concat, length, copy).	6
5	<b>Тема 1.5.</b> Циклы for, while, repeat.	ПЗ 1.5.1. Программы, реализующие работу с циклом for.	2
		ПЗ 1.5.2. Программы, реализующие работу с циклом while...do.	2
		ПЗ 1.5.3. Программы, реализующие работу с циклом repeat...until.	2
		ПЗ 1.5.4. Бесконечные циклы.	2
<b>Раздел 2. Средства для оптимизации программирования</b>			
6	<b>Тема 2.1.</b> Работа со структурированными типами данных.	ПЗ 2.1.1. Программы, реализующие работу с массивами (array).	2
		ПЗ 2.1.2. Программы, реализующие работу с множествами (set).	2
		ПЗ 2.1.3. Программы, реализующие работу с записями (record).	2
7	<b>Тема 2.2.</b> Подпрограммы-процедуры.	ПЗ 2.2.1. Подпрограммы-процедуры. Создание процедуры, обращение к процедуре.	2
		ПЗ 2.2.2. Подпрограммы-процедуры. Формальные и фактические параметры. Параметры-значения, параметры-переменные.	4



№ п/п	Наименование темы	Содержание учебных вопросов	Кол-во часов
8	<b>Тема 2.3.</b> Подпрограммы-функции.	ПЗ 2.3.1. Подпрограммы-функции. Создание функции, обращение к функции.	2
		ПЗ 2.3.2. Подпрограммы-функции. Формальные и фактические параметры. Глобальные и локальные параметры.	4
9	<b>Тема 2.4.</b> Библиотечные модули.	ПЗ 2.4.1. Встроенные библиотечные модули.	2
		ПЗ 2.4.2. Создание библиотечных модулей.	4
10	<b>Тема 2.5.</b> Текстовые и типизированные файлы.	ПЗ 2.5.1. Основы программирования файлов.	2
		ПЗ 2.5.2. Работа с текстовыми файлами.	2
		ПЗ 2.5.3. Работа с типизированными файлами.	2
11	<b>Тема 2.6.</b> Алгоритмы записи в файл и чтения из файла.	ПЗ 2.5.1. Создание блок-схемы алгоритмов записи в файл и чтения из файла.	2
		ПЗ 2.5.2. Программирование созданных блок-схем для текстовых и типизированных файлов.	4
<b>Раздел 3. Работа с динамической памятью</b>			
12	<b>Тема 3.1.</b> Указатели.	ПЗ 3.1.1. Создание указателей. Операции с указателями.	4
13	<b>Тема 3.2.</b> Работа с динамической памятью.	ПЗ 3.2.1. Создание и удаление динамических переменных.	2
		ПЗ 3.2.2. Программирование динамических массивов.	4
14	<b>Тема 3.3.</b> Динамические структуры.	ПЗ 3.3.1. Работа со стеками.	4
		ПЗ 3.3.2. Работа с очередями	4
		ПЗ 3.3.3. Работа с линейными списками	4
15	<b>Тема 3.4.</b> Основы объектно-ориентированного программирования.	ПЗ 3.4.1. Описание объектов. Экземпляры объектов.	2
		ПЗ 3.4.2. Виртуальные методы.	4
		ПЗ 3.4.3. Объекты в динамической памяти.	4
ИТОГО			96

#### 4.2.3. Самостоятельная работа

№ п/п	Тема	Содержание учебных вопросов	Кол-во часов	Формы контроля
1.	<b>Тема 1.1.</b> Принципы построения алгоритмов.	Изучение ГОСТа 19.003-80.	2	Устный опрос.
2.	<b>Тема 1.2.</b> Изучение основ языка программирования.	Алфавит языка Паскаль	2	
3.	<b>Тема 1.3.</b> Ввод-вывод, условные операторы if и case.	Изучение операторов if, case.	4	Проверка задачи.
4.	<b>Тема 1.4.</b> Работа со строковым типом данных.	Работа со строками (delete, insert, concat, length, copy)	4	
5.	<b>Тема 1.5.</b> Циклы for, while, repeat.	Изучение циклов (for, while...do, repeat...until)	4	Проверка задачи.
6.	<b>Тема 2.1.</b> Работа со структурированными типами данных.	Одномерные и двумерные массивы.	4	Проверка задачи.
		Создание собственных типов данных с помощью записей.	4	
		Работа с множествами.	6	Проверка задачи.
7.	<b>Тема 2.2.</b> Подпрограммы-процедуры.	Изучение места и структуры подпрограмм.	6	Устный опрос.
		Написание подпрограмм.	6	Проверка задачи.
8.	<b>Тема 2.4.</b> Библиотечные модули.	Создание собственных библиотек.	4	Проверка задачи.
9.	<b>Тема 2.6.</b> Алгоритмы записи в файл и чтения из файла.	Программирование алгоритма записи в файл и чтения из файла.	6	Проверка задачи.
10.	<b>Тема 3.3.</b> Динамические структуры.	Изучение бинарных деревьев	4	
11.	<b>Тема 3.4.</b> Основы объектно-ориентированного программирования.	Изучение основ объектно-ориентированного программирования	6	Устный опрос
ИТОГО:			<b>62</b>	

## **5. Перечень образовательных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного программного обеспечения, современных профессиональных баз данных и информационных справочных систем**

### **5.1. Образовательные технологии**

При реализации различных видов учебной работы по дисциплине «Основы алгоритмизации и программирования» используются следующие образовательные технологии:

Интерактивные технологии: Лекция «обратной связи» (лекция-беседа)

Инновационные методы, которые предполагают применение информационных образовательных технологий, а также учебно-методических материалов, соответствующих современному мировому уровню, в процессе преподавания дисциплины:

- использование медиаресурсов, энциклопедий, электронных библиотек и Интернет;
- консультирование студентов с использованием электронной почты;
- использование программно-педагогических тестовых заданий для проверки знаний обучающихся.

### **5.2. Лицензионное программное обеспечение**

В образовательном процессе при изучении дисциплины используется следующее лицензионное программное обеспечение:

операционные системы	MS Windows 10 Professional SP1 MS Windows 7 Professional SP1 MS Windows Server 2016 Standard
офисные программы	MS Office 2013 Standart MS Project 2013 Adobe Acrobat 11
базы данных	MS Access 2013
антивирусные пакеты	AVP Kaspersky Endpoint Security 11
система тестирования	INDIGO

### **5.3. Современные профессиональные базы данных**

В образовательном процессе при изучении дисциплины используются следующие современные профессиональные базы данных:

Реферативная и справочная база данных рецензируемой литературы Scopus - <https://www.scopus.com>.

Политематическая реферативно-библиографическая и наукометрическая (библиометрическая) база данных Web of Science - <https://apps.webofknowledge.com>

Архив научных журналов НП Национальный Электронно-Информационный Консорциум (НЭИКОН) ([arch.neicon.ru](http://arch.neicon.ru))

<http://choose-it.ru/article/?id=1237> – информационно-образовательный портал для молодых специалистов ИТ

[http://mirznanii.com/info/informatsionnye-sistemy-i-tekhnologii\\_113221](http://mirznanii.com/info/informatsionnye-sistemy-i-tekhnologii_113221) - Информационные системы и технологии

[bdu.fstec.ru/vul](http://bdu.fstec.ru/vul) – базы данных по угрозам компьютерной безопасности

### **5.4. Информационные справочные системы**

1. СПС «КонсультантПлюс» (договор №СВ16-182).

## **6. Фонд оценочных средств**

Типовые задания, база тестов и иные материалы, необходимые для оценки результатов

освоения дисциплины (в том числе в процессе ее освоения), а также методические материалы, определяющие процедуры этой оценки приводятся в приложении 1 к рабочей программе дисциплины.

Универсальная система оценивания результатов обучения выполняется в соответствии с Положением о формах, периодичности и порядке проведения текущего контроля успеваемости и промежуточной аттестации обучающихся в АНПОО «ККУ», утвержденным приказом директора от 03.02.2020 г. № 31 о/д и включает в себя системы оценок:

- 1) «отлично», «хорошо», «удовлетворительно», «неудовлетворительно»,
- 2) «зачтено», «не зачтено».

При разработке оценочных средств преподавателем используются базы данных педагогических измерительных материалов, предоставленных ООО «Научно-исследовательский институт мониторинга качества образования».

## **7. Основная и дополнительная учебной литература и электронные образовательные ресурсы, необходимые для освоения дисциплины**

### **7.1 Основная учебная литература**

- Долгов, А. И. Алгоритмизация прикладных задач : учебное пособие : [16+] / А. И. Долгов. – Москва : ФЛИНТА, 2021. – 136 с. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=83142> – ISBN 978-5-9765-0086-2. – Текст : электронный.

- Нагаева, И. А. Основы алгоритмизации и программирования : практикум : учебное пособие : [12+] / И. А. Нагаева, И. А. Кузнецов. – Москва ; Берлин : Директ-Медиа, 2021. – 168 с. : схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=598404> (дата обращения: 15.05.2023). – Библиогр.: с. 162-163. – ISBN 978-5-4499-1612-9. – DOI 10.23681/598404. – Текст : электронный.

- Нагаева, И. А. Основы алгоритмизации и программирования : практикум : учебное пособие : [12+] / И. А. Нагаева, И. А. Кузнецов. – Москва ; Берлин : Директ-Медиа, 2021. – 168 с. : схем. – Режим доступа: по подписке. – URL: <https://biblioclub.ru/index.php?page=book&id=598404> (дата обращения: 15.05.2023). – Библиогр.: с. 162-163. – ISBN 978-5-4499-1612-9. – DOI 10.23681/598404. – Текст : электронный.

### **7.2 Дополнительная учебная литература**

- Костюкова, Н.И. Комбинаторные алгоритмы для программистов / Н.И. Костюкова. - 2-е изд./, исправ./ - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 217 с. : ил. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429067>

- Системный администратор (журнал).

### **7.3. Электронные образовательные ресурсы**

Электронно-библиотечная система «Университетская Библиотека Онлайн» - <https://biblioclub.ru/>.

Научная электронная библиотека - [www.elibrary.ru](http://www.elibrary.ru).

Научная библиотека открытого доступа - <https://cyberleninka.ru>

## **8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» (далее - сеть «Интернет»), необходимых для освоения дисциплины**

1. <https://www.ixbt.com/> — специализированный российский информационно-аналитический сайт с самыми актуальными новостями из сферы IT

2. <https://3dnews.ru/> - Интернет издание - публикация новостей и аналитики в компьютерных технологиях, результатов тестирования компьютерной техники (видеокарт, мультимедиа, принтеров, сканеров и др.).
3. <http://www.cnews.ru/> - издание о высоких технологиях. Информация о высоких технологиях.
4. <https://compress.ru/> - Компьютер ПРЕСС – Обзор новостей компьютерной аналитики.
5. <https://www.microsoft.com/ru-ru/learning/training.aspx> /Учебные курсы по ИТ Microsoft
6. <http://www.intuit.ru/> Интернет-университет информационных технологий («ИНТУИТ»)
7. <http://www.elw.ru/> Журнал «e-Learning World – Мир электронного обучения»
8. <https://www.it-world.ru> Новости и аналитика рынка информационных технологий
9. <https://www.osp.ru/> Все новости мира компьютеров и связи.
10. <https://i-exam.ru/> - Единый портал интернет-тестирования в сфере образования

## **9. Требования к минимальному материально-техническому обеспечению, необходимому для осуществления образовательного процесса по дисциплине**

Для изучения дисциплины используется мультимедийная аудитория. Мультимедийная аудитория оснащена современными средствами воспроизведения и визуализации любой видео и аудио информации, получения и передачи электронных документов.

Для изучения дисциплины требуется мультимедийная техника. Специальных материально-технических средств: лабораторного оборудования, компьютерных классов и т.п., для преподавания дисциплины не требуется.

Во время лекционных занятий целесообразно использовать мультимедийную технику, так как практически ко всем лекциям разработаны слайдовые презентации, сопоставительные таблицы и другой материал, который можно продемонстрировать с помощью мультимедийного проектора. В связи с этим материально-техническое обеспечение дисциплины предполагает мультимедийное оборудование. Материально-техническая база должна соответствовать действующим санитарным и противопожарным нормам.

Минимальные требования к оргтехнике:

Процессор: 1,2 ГГц и выше;

Оперативная память: 1 Г и выше;

Другие устройства: Звуковая карта, колонки и/или наушники;

Устройство для чтения DVD-дисков.

Реализация программы дисциплины требует наличия учебных кабинетов: кабинет технических средств информатизации, библиотека, читальный зал с выходом в сеть Интернет.

Для среднего профессионального образования.

Приложение 1  
к рабочей программе дисциплины  
Основы алгоритмизации и  
программирования

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
**Основы алгоритмизации и программирования**

## 1.1. Оценочные средства по итогам освоения дисциплины

### 1.1.1. Цель оценочных средств

**Целью оценочных средств** является установление соответствия уровня подготовленности обучающегося на данном этапе обучения требованиям рабочей программы по дисциплине «Основы алгоритмизации и программирования».

**Оценочные средства** предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «Основы алгоритмизации и программирования». Перечень видов оценочных средств соответствует рабочей программе дисциплины.

**Комплект оценочных средств** включает контрольные материалы для проведения всех видов контроля в форме устного и письменного опроса, практических занятий, и промежуточной аттестации в форме вопросов и заданий к зачету.

**Структура и содержание заданий** – задания разработаны в соответствии с рабочей программой дисциплины «Основы алгоритмизации и программирования».

### 1.1.2. Объекты оценивания – результаты освоения дисциплины

**Объектом оценивания** являются формируемые компетенции ОК1-9, ПК 1.2,1.3,2.2,2.3.

**Результатами освоения** дисциплины являются:

- 3-1 общие принципы построения алгоритмов, основные алгоритмические конструкции;
- 3-2 понятие системы программирования;
- 3-3 основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти;
- 3-4 подпрограммы, составление библиотек программ; объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов
- У-1 использовать языки программирования, строить логически правильные и эффективные программы.

Таблица 1 - Перечень компетенций, формируемых в процессе освоения дисциплины с указанием этапов их формирования

№	Этапы формирования	Перечень компетенций.	Планируемые результаты	Формы
---	--------------------	-----------------------	------------------------	-------

п/п	компетенций в процессе освоения дисциплины. (контролируемые модули, разделы, темы дисциплины (результаты по разделам))	(код контролируемой компетенции (или её части) / и ее формулировка)	освоения дисциплины	контроля, наименования оценочного средства
1	Раздел 1. Основные средства ОАиП	<p>ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.</p> <p>ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.</p> <p>ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность.</p> <p>ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития.</p> <p>ОК 6. Работать в коллективе и в команде, эффективно общаться с коллегами, руководством, потребителями.</p> <p>ОК 7. Брать на себя ответственность за работу членов команды (подчиненных), за результат выполнения заданий.</p> <p>ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.</p>	<p>3-1 общие принципы построения алгоритмов, основные алгоритмические конструкции;</p> <p>3-2 понятие системы программирования;</p> <p>3-3 основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти</p>	Входной контроль (контрольная работа)
2	Раздел 2. Средства для оптимизации программирования	<p>. ОК 5. Использовать информационно-коммуникационные технологии в профессиональной деятельности.</p> <p>ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности.</p>	3-4 подпрограммы, составление библиотек программ; объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов	Текущий (контроль (тест))
3	Раздел 3. Работа с динамической памятью.	ПК1.2 Взаимодействовать со специалистами смежного профиля при разработке	У-1 использовать языки программирования, строить логически	Рубежные контроль (задания )

	методов, средств и технологий применения объектов профессиональной деятельности. ПК 1.3 Производить модификацию отдельных модулей информационной системы в соответствии с рабочим заданием, документировать произведенные изменения. ПК2.2. Программировать в соответствии с требованиями технического задания. ПК 2.3. Применять методики тестирования разрабатываемых приложений. ПК 2.4. Формировать отчетную документацию по результатам работ	правильные и эффективные программы.	ПА (экзамен)
--	--	-------------------------------------	--------------

### 1.1.3. Формы контроля и оценки результатов освоения

Контроль и оценка результатов освоения – это выявление, измерение и оценивание знаний и умений формирующихся компетенций в рамках освоения дисциплины. В соответствии с учебным планом и рабочей программой дисциплины «Основы алгоритмизации и программирования» предусматривается входной, текущий, рубежный и итоговый контроль результатов освоения (промежуточная аттестация в форме экзамена).

### 1.2. Примерные ( типовые ) контрольные задания или иные материалы, необходимые для оценки знаний, умений, владений (или опыта деятельности), в процессе освоения дисциплины (модуля, практики), характеризующих этапы формирования компетенций в процессе освоения дисциплины

#### 1.2.1. Примерные ( типовые ) контрольные задания или иные материалы для проведения входного контроля

##### Контрольная работа (на уровне знаний)

1. Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе?  
Какова структура ячейки памяти?
2. Что такое конъюнкция? Создайте таблицу истинности для конъюнкции.  
Сформулируйте 2 правила получения дополнительного кода числа.
3. Что такое дизъюнкция? Создайте таблицу истинности для дизъюнкции.  
Сложить и проверить: 011010010+10101011
4. Что такое импликация? Создайте таблицу истинности для импликации.  
Перевести из шестнадцатеричной системы счисления в десятичную: 1F8
5. Создайте таблицу истинности для отрицания.  
Вычесть и проверить: 11011000 – 01100110
6. Какие высказывания являются эквивалентными? Создайте таблицу истинности для эквивалентности.  
Пусть n – число линий в шине адреса. Каков объем адресного пространства?
7. Перечислите свойства алгоритма.  
Перевести из десятичной системы счисления в шестнадцатеричную: 730
8. Назовите способы представления алгоритма.  
Сложить и проверить: 10D+18



9. Как графически представить элемент алгоритма «проверка условия»?  
Перевести из шестнадцатеричной системы счисления в десятичную: 7BC
10. Сколько байт содержится в слове, двойном слове, параграфе?  
Сложить и проверить: 11110001+01010101
11. Назовите критерии качества программного изделия.  
Вычсть и проверить: 11000110-00111100
12. Назовите атрибуты файлов на дисках.  
Сложить и проверить: CCA+ADB.
13. Какие языки программирования высокого уровня вы знаете?  
Вычсть и проверить: F83 – 3B.
14. Как называется язык программирования низкого уровня?  
Перевести из десятичной системы счисления в шестнадцатеричную: 640.
15. Что такое трансляция программы?  
Перевести из двоичной системы счисления в десятичную: 01010010
16. Что такое система программирования?  
Перевести из шестнадцатеричной системы счисления в десятичную: 8AE
17. Как графически изображается ввод/вывод данных?  
Сложить и проверить: C1A+A0B.
18. Назовите свойства ООП.  
Вычсть и проверить: 7C - 6F.
19. Как графически изображается обработка данных?  
Сложить и проверить: 01010010+01110011
20. Какую операцию в блок-схеме обозначает ромб?  
Перевести из шестнадцатеричной системы счисления в десятичную: 3E6
21. Какую операцию в блок-схеме обозначает прямоугольник?  
Перевести из двоичной системы счисления в десятичную: 01110011
22. Какую операцию в блок-схеме обозначает параллелограмм?  
Перевести из десятичной системы счисления в шестнадцатеричную 477
23. Перечислите свойства алгоритма.  
Какова структура ячейки памяти?
24. Что такое система программирования?  
Сформулируйте 2 правила получения дополнительного кода числа.
25. Назовите критерии качества программного изделия.  
Назовите свойства ООП.
26. Что такое конъюнкция? Создайте таблицу истинности для конъюнкции.  
Сложить и проверить: C1A+A0B.
27. Что такое компиляция программы?  
Что такое дизъюнкция? Создайте таблицу истинности для дизъюнкции
28. Как называется язык программирования низкого уровня?  
Сложить и проверить: 011010010+10101011
29. Как графически представить элемент алгоритма «проверка условия»?  
Пусть n – число линий в шине адреса. Каков объем адресного пространства?
30. Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе?  
Какие высказывания являются эквивалентными? Создайте таблицу истинности для эквивалентности.

### **1.2.2. Примерные (типовые) контрольные задания или иные материалы для проведения текущего контроля**

#### **Тест (ОК 1, 5, ПК 1.2, 1.3, 2.2, 2.3) (на уровне знаний)**

1. Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе? Что такое библиотечный модуль? Какова его структура?
2. Каков диапазон значений и объем памяти типа integer? word? byte? Должно ли совпадать имя библиотечного модуля с именем файла, который его содержит, или это необязательно? Как обратиться к библиотечному модулю в теле головной программы?

3. Каков объем памяти типов string и char? В чем различие между типами string и char? Какова максимально возможная длина строки? Какое соответствие должно соблюдаться между формальными и фактическими параметрами в подпрограммах?
4. В чем сходство и отличие между типами byte, word, integer? Расскажите об операциях, которые можно выполнять над строковыми величинами. Какая функция (процедура) является аналогом операции сцепления (+) при работе со строками?
5. Где в программе объявляются переменные? Как объявляется логический тип? Какие он может принимать значения? Что такое автоматический контроль ввода-вывода при работе с файлами? Как его отключить?
6. Покажите общую структуру программы на языке PASCAL. Может ли фактических параметров процедуры быть больше, чем формальных? А меньше?
7. Как работают операторы ввода READ, READLN? Чем они отличаются друг от друга? Что такое подпрограмма? Какие параметры подпрограммы называют формальными? фактическими?
8. Как работают операторы вывода WRITE, WRITELN? Чем они отличаются друг от друга? Что такое массив? Что такое размерность массива? Объявите в блоке VAR одномерный массив с именем MAS, состоящий из десяти элементов целого типа.
9. Когда возникает необходимость в условном операторе? Как называется этот оператор и каков формат его записи? Каков формат объявления данных (покажите на примере)? Какие типы данных вы знаете?
10. Расскажите об операторе выбора case. В чем его сходство с условным оператором? Какого типа должен быть параметр счетного цикла FOR? Покажите на примере.
11. Чем полный условный оператор отличается от неполного условного оператора? Какова структура счетного цикла FOR?
12. Какова структура циклов с пред- и постусловием (WHILE ... DO и REPEAT ... UNTIL)? Как выполняются эти циклы? Каков объем памяти типа real? Как еще можно объявить действительные числа?
13. Сколько раз исполнится фрагмент программы? For i := 1 to 10 Do k:=k+2; С каким шагом здесь изменяется переменная цикла? Укажите ее. Чем отличается инициализация статического и динамического массивов?
14. Может ли фактических параметров процедуры быть больше, чем формальных? А меньше? Как работает команда eof ()? Какой тип она возвращает?
15. Что такое библиотечный модуль? Какова его структура? Должно ли совпадать имя библиотечного модуля с именем файла, который его содержит, или это необязательно? Как обратиться к библиотечному модулю в теле головной программы?
16. Когда возникает необходимость в операторе выбора? Как называется этот оператор и каков формат его записи? Что такое массив? Что такое размерность массива? Объявите в блоке VAR двумерный массив с именем MAS, состоящий из 3-х строк и 4-х столбцов целого типа.
17. Что такое подпрограмма? Какие параметры подпрограммы называют формальными? фактическими? Каков диапазон значений и объем памяти типа integer? word? byte?
18. Как объявить файловую переменную? Зачем при программировании файлов необходимо отключать автоматический контроль ввода/вывода? Что такое блоки var, const?
19. Что обозначает {\$I-}? Как создать собственный тип данных?
20. Для чего используется оператор type? Где применяется команда Assign?
21. Чем команда Append отличается от Rewrite? Какая информация записывается в блоки Interface и Implementation в библиотечном модуле?
22. Как инициализировать файл на запись? Можно ли в программе подключить несколько библиотек?
23. Как инициализировать файл на чтение? Какое расширение у файла, содержащего исходный текст программы? Исходный текст библиотечного модуля?
24. Какой оператор проверяет существование файла? Что такое компиляция программы?
25. Что делает команда Reset? Какое расширение у файла, содержащего откомпилированный текст программы? Откомпилированный текст библиотечного модуля?
26. Какая команда закрывает файл? Почему файл обязательно необходимо закрыть? Какова структура программы на языке PASCAL?

- 27 Для чего необходима файловая переменная при работе с файлами? Как работают вложенные циклы? Приведите пример.
- 28 Покажите программно, как записать строку в файл? Что такое идентификатор? Каким условиям должен удовлетворять идентификатор, чтобы не было ошибки в программе?
- 29 Как прочитать строку из файла? Как прочитать все строки, находящиеся в файле? Должны ли совпадать тип и количество формальных и фактических параметров в подпрограмме?
- 30 Для чего нужна команда Ioresult? Приведите пример ее использования. Где объявляются подпрограммы в Turbo Pascal? Существуют ли подпрограммы без параметров?

### Критерии оценивания теста

% правильных ответов	Оценка по традиционной системе
85-100	отлично
70-84	хорошо
50-69	удовлетворительно
0-49	неудовлетворительно

### 1.2.3. Примерные ( типовые ) контрольные задания или иные материалы для проведения рубежного контроля (ОК 1,5,ПК 1.1,1.,2.2,2.3) (на уровне умения)

№	Содержание задания	Порядок выполнения работы:
<b>Тема: Массивы и оператор цикла</b>		
1	Какой тип могут иметь индексы массивов?	<i>Спроектировать и реализовать на Паскале программу со следующими программными функциями:</i> 1.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает минимальный элемент во введенном массиве, выводит его значение и значения его индексов; 2.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает максимальный элемент во введенном массиве; 3.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает максимальный элемент главной диагонали введенного массива, выводит его значение и значения его индексов; 4.Программа запрашивает и вводит элементы массива с клавиатуры терминала, отыскивает максимальный элемент побочной диагонали введенного массива, выводит его значение и значения его индексов.
2	Какого типа могут быть элементы массивов?	
3	Массивы какой размерности допустимы в Паскале?	
4	Как можно в программе сослаться на элемент массива?	
5	Управляющие переменные какого типа допустимы в операторе for?	
6	Какого типа выражения допустимы в качестве начального и конечного значений?	
7	Как необходимо записать несколько операторов, чтобы они выполнялись в цикле for?	
<b>Тема: Процедуры Паскаля</b>		
1	Как можно передавать данные из вызывающей программы в процедуру?	1.Напишите процедуру вычисления квадратов двух дробных чисел и вывода на экран
2	Как можно передать результаты выполнения процедуры в	

	вызывающую её программу?	их суммы;
3 3	Какие два типа формальных параметров определены для процедуры?	2. Напишите процедуру вычисления суммы квадратов двух дробных чисел и их разности;  Программа должна работать в режиме диалога. В варианте 1 опишите процедуру с двумя параметрами-значениями. Результат печатайте в самой процедуре. В варианте 2 опишите процедуру с двумя параметрами-значениями и двумя параметрами-переменными. Результат печатайте в головной программе.

### 1.2.4. Примерные ( типовые ) контрольные задания или иные материалы для проведения промежуточной аттестации

#### Примерные ( типовые ) вопросы к экзамену по дисциплине «Основы алгоритмизации и программирования»

##### 1. на уровне знаний

Вопрос	Код компетенции (согласно РПД)
1. Алгоритм и его свойства. Графическое изображение алгоритмов линейной, разветвляющейся и циклической структур. Основные требования ГОСТа 19.003-80..	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
1. Система программирования.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
2. Характеристика языка Турбо Паскаль. Алфавит языка Паскаль.	ОК 1-9, ПК 1.2,1. ОК 1-9, ПК 1.2,1.3,2.2,2.3,3.2,2.2,2.3.
3. Структура программы на Паскале. Простые типы данных. Краткая характеристика.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
4. Операции и выражения. Оператор присваивания. Составной оператор.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
5. Операторы ввода-вывода WRITE, READ. Инициализация переменных.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
6. Строки. Операции над строками. Стандартные процедуры и функции для строк.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
7. Условный оператор <i>if</i> . Оператор выбора <i>case</i> .	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
8. Операторы цикла (счетный, с предусловием, с постусловием).	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
9. Множества. Операции над множествами.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
10. Метки и операторы перехода.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
11. Записи. Оператор присоединения With ... do.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
12. Структурированные типы данных. Краткая характеристика.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
13. Подпрограммы-процедуры. Обращение к процедуре. Параметры-значения. Параметры-переменные.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
14. Подпрограммы-функции. Обращение к функции. Формальные и фактические параметры.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
15. Одномерные массивы. Методы сортировки массивов.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
16. Одномерные массивы. Алгоритм поиска минимального и максимального элементов.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.

17.	Многомерные массивы. Методы работы с двумерными массивами.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
18.	Процедуры работы с текстовыми файлами.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
19.	Текстовые файлы. Алгоритм записи данных в файл. Ошибки открытия файла	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
20.	Текстовые файлы. Алгоритм чтения данных из файла. Ошибки открытия файла.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
21.	Создание библиотечного модуля.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.
22.	Интерфейсная, исполняемая и иницирующая части модуля. Компиляция модулей.	ОК 1-9, ПК 1.2,1.3,2.2,2.3.

**Примерные (типовые) задания (оценочные средства), выносимые на экзамен  
2. на уровне умений**

Содержание задания	Код компетенции (согласно РПД)
Вычислить значение выражения $y = x * 1/4$ ; где $x, n$ – задаются пользователем с клавиатуры. Результаты вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Определить сколько положительных, четных значений попадает в промежуток от -12 до +12. Результат вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Организовать диалог пользователя и компьютера по заполнению паспортных данных.	ПК 1.2,1.3,2.2,2.3.
Составить программу вычислений следующих функций: Результаты вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Определить, какое из трех значений, вводимых пользователем с клавиатуры, является кратным 5. Вывести на экран соответствующее сообщение.	ПК 1.2,1.3,2.2,2.3.
Определить количество отрицательных значений из 15 чисел, вводимых пользователем с клавиатуры. Результат вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Определить разность всех четных чисел, вводимых пользователем с клавиатуры. Ввод данных производить до тех пор, пока не встретится число 0. Результаты вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Вычислить где $n$ изменяется от 10 до 0 с шагом 2. Значение $x$ вводится пользователем с клавиатуры. Результаты вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Вычислить значение выражения $y = x * 1/3$ ; где $x, n$ – задаются пользователем с клавиатуры. Результаты вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Определить сколько положительных значений, кратных 3 попадает в промежуток от -15 до +15. Результат вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Найти среднеарифметическое значение чисел от 10 до 25. Результат вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный целочисленный массив, размером $N$ , числами. Заменить все числа кратные 8 на 0. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.
Заполнить двумерный целочисленный массив, размером $N$ , числами. Зеркально отобразить его содержимое по горизонтали. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.
Заполнить двумерный массив случайными числами. Зеркально отобразить его содержимое по вертикали. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.
Заполнить двумерный массив случайными числами. Повернуть его содержимое на 90 градусов. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.

Заполнить двумерный массив случайными числами. Повернуть его содержимое на 180 градусов. Вывести массив на экран до и после модификации	ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный массив случайными числами в диапазоне от 0 до 50. Определить среднее арифметическое всех чисел входящих в массив и записать это значение в последний элемент массива. Вывести массив на экран до и после модификации	ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный массив случайными числами. Посчитать количество элементов между максимальным и минимальным элементами массива. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный массив случайными числами. Поменять местами максимальный и минимальный элементы массива. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный массив случайными числами. Используя побитовые операции прибавить ко всем четным числам 1. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный массив размерностью > 30 элементов случайными целыми числами из диапазона от 10 до 20. Создать второй массив и перенести в него элементы первого массива, но так, чтобы они встречались в нем только один раз. Вывести оба массива на экран.	ПК 1.2, ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный массив размерностью > 30 элементов случайными целыми числами из диапазона от 10 до 20. Посчитать какое количество элементов встречается в массиве больше 2 раз. Вывести массив и результат на экран.	П ПК 1.2,1.3,2.2,2.3.К 1.2,1.3,2.2,2.3.
Заполнить одномерный массив вещественными числами по формуле $x = \sin(i)$ , где $x$ номер элемента массива. Вывести массив на экран. Заполнить одномерный массив вещественными числами по формуле $x = \cos(i)$ , где $A, B, C$ день и месяц сдачи экзамена и номер билета соответственно, а $x$ номер элемента массива. Вывести массив на экран.	ПК 1.2,1.3,2.2,2.3.
Дан двумерный целочисленный массив $A$ размером $4 \times 4$ . Заполнить массив числами. Определить максимальный элемент массива. Вывести на экран полученный массив и значение максимального элемента.	ПК 1.2,1.3,2.2,2.3.
Дан двумерный целочисленный массив $A$ размером $4 \times 4$ . Заполнить массив числами. Определить минимальный элемент массива. Вывести на экран полученный массив и значение минимального элемента.	ПК 1.2,1.3,2.2,2.3.
Написать программу, которая запрашивает десятичное число и вычисляет сумму цифр четных позиций.	ПК 1.2,1.3,2.2,2.3.
Заполнить одномерный массив случайными числами. Используя побитовые операции прибавить ко всем нечетным числам 1. Вывести массив на экран до и после модификации.	ПК 1.2,1.3,2.2,2.3.
Написать программу, которая вычисляет сумму цифр натурального числа, введенного с клавиатуры.	ПК 1.2,1.3,2.2,2.3.
Дана целочисленная прямоугольная матрица. Вычислить количество строк, содержащих хотя бы один нулевой элемент.	ПК 1.2,1.3,2.2,2.3.
Написать программу, которая запрашивает две строки, а затем соединяет их и выводит новую строку на экран.	ПК 1.2,1.3,2.2,2.3.
Написать программу, которая запрашивает десятичное число и вычисляет сумму цифр нечетных позиций.	ПК 1.2,1.3,2.2,2.3.
Заполнить два одномерных массива случайными числами. Вывести на экран оба массива и числа присутствующие в обоих массивах одновременно.	ПК 1.2,1.3,2.2,2.3.
Дана целочисленная прямоугольная матрица. Вычислить количество положительных элементов по каждому столбцу.	ПК 1.2,1.3,2.2,2.3.
Написать программу, которая считывает текст из файла и выводит на экран количество восклицательных предложений.	ПК 1.2,1.3,2.2,2.3.
Составить функцию определения простого числа. Используя функцию определить количество простых чисел в массиве.	ПК 1.2,1.3,2.2,2.3.
Даны две даты в формате ДД.ММ.ГГ. Написать функцию определяющую наибольшее из них. Формат даты задать в виде структуры.	ПК 1.2,1.3,2.2,2.3.
Написать функцию, определяющую количество счастливых билетов в рулоне проездных билетов. Считать, что рулон начинается с билета № 423445 и содержит 200 билетов. Результат вывести на экран.	ПК 1.2,1.3,2.2,2.3.
Составить банк данных о маршрутах поездов, указав: номер поезда, время	ПК 1.2,1.3,2.2,2.3.

отправления, время в пути, промежуточные пункты маршрута. Данные записать в файл.	
Составить банк данных о маршрутах поездов, указав: номер поезда, время отправления, время в пути, промежуточные пункты маршрута. Данные записать в файл.	ПК 1.2,1.3,2.2,2.3.
Записать в файл последовательного доступа N целых чисел, полученных с помощью генератора случайных чисел. Подсчитать количество пар противоположных чисел среди компонентов этого файла.	ПК 1.2,1.3,2.2,2.3.
Заполнить файл последовательного доступа f целыми числами, полученными с помощью генератора случайных чисел. Получить в файле g те компоненты файла f, которые являются четными.	ПК 1.2,1.3,2.2,2.3.
Построить описание класса содержащего информацию о почтовом адресе организации. Предусмотреть возможность раздельного изменения составных частей адреса, создания и уничтожения объектов этого класса.	ПК 1.2,1.3,2.2,2.3.
Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменения размеров.	П ПК 1.2,1.3,2.2,2.3.К 1.2,1.3,2.2,2.3.
Составить описание класса для объектов-векторов, задаваемых координатами концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами.	П ПК 1.2,1.3,2.2,2.3.К 1.2,1.3,2.2,2.3.
Составить описание класса для работы с цепными списками строк (строки произвольной длины) с операциями включения в список, удаления из списка элемента с заданным значением данного.	ПК 1.2,1.3,2.2,2.3.
Построить систему классов для описания плоских геометрических фигур: круга, квадрата, прямоугольника. Предусмотреть методы для создания объектов, перемещения на плоскости.	ПК 1.2,1.3,2.2,2.3.

## **Методические рекомендации и указания**

### **1. Методические указания для обучающихся по освоению дисциплины**

Дисциплина «Основы алгоритмизации и программирования» считается освоенной обучающимся, если он имеет положительные результаты входного, текущего контроля и промежуточной аттестации.

Для достижения вышеуказанного обучающийся должен соблюдать следующие правила, позволяющие освоить дисциплину на высоком уровне:

1. Начало освоения курса должно быть связано с изучением всех компонентов программы дисциплины «Основы алгоритмизации и программирования» с целью понимания ее содержания и указаний, которые будут доведены до сведения обучающегося на первой лекции и первом занятии семинарского типа.

Перед началом курса целесообразно ознакомиться со структурой дисциплины на основании программы, а так же с последовательностью изучения тем и их объемом. С целью оптимальной самоорганизации необходимо сопоставить эту информацию с графиком занятий и выявить наиболее затратные по времени и объему темы, чтобы заранее определить для себя периоды объемных заданий.

2. Каждая тема содержит лекционный материал, список литературы для самостоятельного изучения, вопросы и задания для подготовки к занятиям семинарского типа. Необходимо заранее обеспечить себя этими материалами и литературой или доступом к ним.

3. После лекции необходимо изучить лекционный материал по соответствующей теме, обратить особое внимание на актуальные и проблемные вопросы рассматриваемой темы.

4. Занятие семинарского типа, как правило, начинается с опроса по лекционному материалу темы и материалам указанных к теме литературных источников. В связи с этим подготовка к практическому занятию заключается в повторении лекционного материала и изучении вопросов предстоящего занятия.

При возникновении затруднений с пониманием материала занятия обучающийся должен обратиться с вопросом к преподавателю для получения соответствующих



разъяснений в отведенное для этого преподавателем время на занятии либо по электронной почте. В интересах обучающегося своевременно довести до сведения преподавателя информацию о своих затруднениях в освоении предмета и получить необходимые разъяснения.

5. Подготовка к экзамену является заключительным этапом изучения дисциплины. Экзамен проводится в устной форме. Каждый билет содержит по два вопроса: один – теоретический, второй – практическое задание (или тесты).

Содержание вопросов находится в доступном режиме с начала изучения дисциплины. В связи с этим целесообразно изучать вопросы не в период экзаменационной сессии непосредственно в дни перед зачетом, а по каждой теме вместе с подготовкой к соответствующему текущему занятию. Кроме того необходимо помнить, что часть вопросов (не более 10%) непосредственно перед экзаменом может быть дополнена или изменена. В связи с этим целесообразно изучать не только вопросы, выносимые на экзамен, но и иные вопросы, рассматриваемые на лекциях и занятиях семинарского типа.

## **2. Методические указания по подготовке к сдаче экзамена**

Экзамен является итоговой формой контроля знаний обучающегося по ««Основы алгоритмизации и программирования», способом оценки результатов его учебной деятельности. Основной целью экзамена является проверка степени усвоения полученных обучающимся знаний и их системы.

Для успешной сдачи зачета необходимо продемонстрировать разумное сочетание знания и понимания учебного материала. На экзамене проверяется не столько механическое запоминание обучающимся изложенной информации, сколько его способность её анализировать, объяснять, аргументировать и отстаивать свою позицию.

К экзамену целесообразно готовиться с самого начала учебного цикла, поскольку только систематическая подготовка может обеспечить формирование у обучающегося качественных системных знаний.

При подготовке следует пользоваться комплексом различных источников - не только конспектами лекций, материалами по подготовке к семинарским занятиям, но также и учебной, научной, справочной литературой.

Преподаватель вправе задать на экзамене обучающемуся наводящие, уточняющие и дополнительные вопросы в рамках билета.

Основными критериями, которыми преподаватель руководствуется при оценке знаний, являются следующие:

- соответствие ответа обучающегося теме вопросов;
- умение строить ответ полно, но лаконично с акцентом на наиболее важных моментах;
- степень осведомлённости о научных и нормативных источниках;
- умение связывать теорию с практикой;
- приведение конкретных примеров.

## **Рекомендации по проведению учебных занятий с обучающимися с применением электронного обучения и дистанционных образовательных технологий**

Для проведения контактной работы обучающихся с преподавателем АНПОО «ККУ» с применением электронного обучения и дистанционных образовательных технологий определен набор электронных ресурсов и приложений, которые рекомендуются к использованию в образовательном процессе. Образовательный процесс осуществляется в соответствии с расписанием учебных занятий 2023/2024 учебного года, размещенным на официальном сайте колледжа.

Организация образовательного процесса осуществляется через личный кабинет на официальном сайте колледжа. Преподаватель в электронном журнале для соответствующей учебной группы указывает тему занятия. Прикрепляет учебные материалы, задания или ссылки на электронные ресурсы, необходимые для освоения темы, выполнения домашних заданий.

Алгоритм дистанционного взаимодействия:

1.1. Для обеспечения дистанционной связи с обучающимися преподаватель взаимодействует с обучающимися групп в электронной платформе Сферум, либо посредством корпоративной электронной почты (домен @kiu39.ru/ @kku39.ru).

1.2. В сформированных группах обучающихся на платформах (см. выше) преподаватель доводит до обучающихся информацию:

- об алгоритме размещения информации об учебных материалах и заданиях на электронных ресурсах колледжа.

- индивидуальный график консультирования обучающихся, в т.ч. дистанционном формате.

1.3. Обучающиеся выполняют задание, в соответствии с расписанием учебных занятий в формате ДО и предоставляют их в электронной форме на электронный ресурс.

1.4. Осуществление мониторинга выполнения учебного плана и посещаемости занятий происходит ежедневно преподавателем через электронные ресурсы.