

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»

Структурное подразделение «Автоматизации и управления»

УТВЕРЖДЕНА:
на заседании кафедры
Протокол №11 от 11 февраля 2025 г.

Рабочая программа дисциплины

«ПРОГРАММИРОВАНИЕ И АЛГОРИТМИЗАЦИЯ»

Направление: 15.03.04 Автоматизация технологических процессов и производств

Системы и средства автоматизации в металлургической промышленности

Квалификация: Бакалавр

Форма обучения: очная

Документ подписан простой электронной подписью Составитель программы: Колодин Алексей Александрович Дата подписания: 18.09.2025

Документ подписан простой электронной подписью Утвердил и согласовал: Елшин Виктор Владимирович Дата подписания: 22.09.2025

Год набора – 2025

Иркутск, 2025 г.

1 Перечень планируемых результатов обучения по дисциплине, соотнесённых с планируемыми результатами освоения образовательной программы

1.1 Дисциплина «Программирование и алгоритмизация» обеспечивает формирование следующих компетенций с учётом индикаторов их достижения

Код, наименование компетенции	Код индикатора компетенции
ОПК ОС-8 Способность разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК ОС-8.1

1.2 В результате освоения дисциплины у обучающихся должны быть сформированы

Код индикатора	Содержание индикатора	Результат обучения
ОПК ОС-8.1	Способен программировать на языке низкого уровня процессы обработки информации и управления техническими системами, владеет техникой объектно-ориентированного программирования	Знать - об основных видах, этапах проектирования программных продуктов; - структурное и модульное программирование; - о методах и средствах объектно-ориентированного программирования. Уметь - сформулировать задачу; - разработать алгоритм; - создать программу и отладить ее; - использовать компьютер для получения, накопления, хранения и обработки данных, представляя их в наглядной форме Владеть - методами проектирования программных алгоритмов; - навыком выбора методов решения задач и разработки алгоритмов; - методами сортировки и поиска; - синтаксисом и семантикой алгоритмического языка программирования C++; - принципами и практиками объектно-ориентированного программирования;

2 Место дисциплины в структуре ООП

Изучение дисциплины «Программирование и алгоритмизация» базируется на результатах освоения следующих дисциплин/практик: «Информационные технологии», «Математика»

Дисциплина является предшествующей для дисциплин/практик: «Информационные технологии в проектировании и управлении», «Проектирование автоматизированных систем», «ПЛК в системах управления», «Программирование

промышленных контроллеров», «Программное обеспечение АСУ ТП», «Интегрированные системы проектирования и управления», «Распределенные системы управления технологическими процессами и производствами»

3 Объем дисциплины

Объем дисциплины составляет – 4 ЗЕТ

Вид учебной работы	Трудоемкость в академических часах (Один академический час соответствует 45 минутам астрономического часа)	
	Всего	Семестр № 3
Общая трудоемкость дисциплины	144	144
Аудиторные занятия, в том числе:	60	60
лекции	20	20
лабораторные работы	0	0
практические/семинарские занятия	40	40
Самостоятельная работа (в т.ч. курсовое проектирование)	48	48
Трудоемкость промежуточной аттестации	36	36
Вид промежуточной аттестации (итогового контроля по дисциплине)	Экзамен	Экзамен

4 Структура и содержание дисциплины

4.1 Сводные данные по содержанию дисциплины

Семестр № 3

№ п/п	Наименование раздела и темы дисциплины	Виды контактной работы						СРС		Форма текущего контроля
		Лекции		ЛР		ПЗ(СЕМ)				
		№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	
1	2	3	4	5	6	7	8	9	10	11
1	Введение	1	1							
2	Ввод-вывод	8	2			9	2	1, 2	4	Отчет
3	Динамическая память	9	2			10	4	1, 2	4	Отчет
4	Модули	10	2			11	2	1, 2	4	Отчет
5	Объекты	11	2			12	4	1, 2	4	Отчет
6	Синтаксис и семантика алгоритмического языка C++	3, 4	3			1, 2, 3, 4, 5	14	1, 2	14	Контрольн ая работа, Отчет
7	Проектирование программных продуктов	2	1					2	2	Отчет
8	Представление набора данных строкой и множеством	5	1			6	2	1, 2	4	Отчет
9	Процедуры и функции	7	2			8	4	1, 2	4	Отчет
10	Сложные структуры	12	2			13	2	1, 2	4	Отчет

	данных									
11	Сортировка и поиск	6	2			7	6	1, 2	4	Отчет
	Промежуточная аттестация								36	Экзамен
	Всего		20				40		84	

4.2 Краткое содержание разделов и тем занятий

Семестр № 3

№	Тема	Краткое содержание
1	Введение	Программирование. Основные виды, этапы проектирования и жизненный цикл программных продуктов
2	Ввод-вывод	Процедуры ввода-вывода в C++. Файлы. Задание файлового типа. Буферная переменная и ее использование. Текстовые файлы. Управление размещением данных по строкам и по позициям в строке. Типизированные и нетипизированные файлы.
3	Динамическая память	Адреса и указатели. Объявление указателей. Выделение и освобождение динамической памяти. Использование указателей. Процедуры и функции для работы с динамической памятью.
4	Модули	Механизм модулей. Состав и структура модулей. Заголовок модуля и связь модулей друг с другом. Интерфейсная часть. Исполняемая часть. Иницилирующая часть. Компиляция модулей. Доступ к объявленным в модуле объектам. Стандартные модули.
5	Объекты	Идея объектно-ориентированного программирования. Инкапсуляция. Наследование. Полиморфизм. Динамические объекты. Экспорт объектов.
6	Синтаксис и семантика алгоритмического языка C++	Алфавит языка. Классификация типов данных. Простые типы данных и простые операторы языка. Выражения. Структура программы. Структурированные операторы языка. Структурированные типы данных. Массивы. Матрица, описание матрицы, работа с матрицами.
7	Проектирование программных продуктов	Основные принципы и подходы. Классы алгоритмов.
8	Представление набора данных строкой и множеством	Строки и операции над ними. Множества и действия над ними.
9	Процедуры и функции	Описание процедуры и функции. Процедуры без параметров и с параметрами. Параметры-значения и параметры – переменные. Вызов процедуры и функции. Рекурсивные функции. Опережающее описание. Процедурные типы. Нетипизированные

		параметры переменные. Рекурсия и итерация.
10	Сложные структуры данных	Ссылки. Списки. Деревья. Двоичные деревья. Деревья общего вида.
11	Сортировка и поиск	Сортировка выбором, обменом, простыми вставками.

4.3 Перечень лабораторных работ

Лабораторных работ не предусмотрено

4.4 Перечень практических занятий

Семестр № 3

№	Темы практических (семинарских) занятий	Кол-во академических часов
1	Простые типы данных, простые операторы языка, выражения. Структура программы	2
2	Составные операторы. Условный оператор	2
3	Счетный оператор цикла for, оператор цикла с предусловием, оператор цикла с постусловием, оператор выбора	2
4	Массивы. Описание вектора. Работа с одномерным массивом: вычисление суммы, среднего арифметического и произведения элементов массива, поиск минимального и максимального элементов массива и их положения в нем, подсчет числа вхождений в массив заданных символов. Линейный поиск в массиве	4
5	N-мерные массивы. Матрица, описание. Работа с матрицами: вычисление суммы, среднего арифметического и произведения элементов матрицы, поиск минимального и максимального элементов матрицы и их положения в ней, подсчет числа вхождений в матрицу заданных символов. Вставка и удаление строк и столбцов	4
6	Строки и операции над ними	2
7	Сортировки: выбором, обменом, простыми вставками	6
8	Подпрограммы. Способы подстановки параметров. Рекурсивные подпрограммы. Процедурный тип	4
9	Представление файлов в программе и операции с ними. Операции ввода – вывода	2
10	Адреса и указатели. Объявление указателей. Выделение и освобождение динамической памяти. Использование указателей	4
11	Механизм модулей. Состав и структура модулей. Заголовок модуля и связь модулей друг с другом	2

12	Динамические объекты. Экспорт объектов	4
13	Ссылки. Списки. Деревья. Двоичные деревья. Деревья общего вида	2

4.5 Самостоятельная работа

Семестр № 3

№	Вид СРС	Кол-во академических часов
1	Оформление отчетов по лабораторным и практическим работам	24
2	Проработка разделов теоретического материала	24

В ходе проведения занятий по дисциплине используются следующие интерактивные методы обучения: Компьютерные симуляции, интерактивная (проблемная) лекция

5 Перечень учебно-методического обеспечения дисциплины

5.1 Методические указания для обучающихся по освоению дисциплины

5.1.1 Методические указания для обучающихся по практическим занятиям

Практические занятия предназначены для закрепления полученных теоретических знаний и решения практических задач.

На практических занятиях студенту необходимо изучить дополнительный материал по дисциплине, не вошедший в основной лекционный курс и последовательно выполнить работы, приведенные в настоящих методических указаниях, оформление и защита отчета. В каждом задании студенту необходимо сделать математическое описание, составить блок-схему алгоритма, написать исходный, откомпилировать проект, отловить ошибки, вывести итоговый результат работы. При выполнении работы должен быть проведен анализ различных вариантов, результатом которого будет грамотное принятие решения. Задания выполняются в компьютерных залах, при выполнении необходимо использовать установленное программное обеспечение.

Все практические работы выполняются в следующем порядке:

- повторяется (изучается) лекционный материал согласно теме работы;
- осуществляется выбор математических решений;
- разрабатывается алгоритм
- кодируется и компилируется программа
- анализируется программа на ошибки
- вывод итогового результата
- подготавливается отчет о проведенной работе

5.1.2 Методические указания для обучающихся по самостоятельной работе:

Цель самостоятельной работы – закрепление и углубление теоретического материала по дисциплине программирование и основы алгоритмизации.

Перечень рекомендуемых самостоятельных работ:

- проработка лекционного материала
- подготовка к промежуточному контролю знаний
- подготовка к экзамену
- подготовка к зачету

- решение задач
- подготовка к лабораторным и практическим работам

Задания на самостоятельную работу.

При подготовке к контрольной работе необходимо проработать изученный материал практических заданий. Внимательно проследить ход решения задач. Попробовать самостоятельно прорешать аналогичные задания из литературы.

В практических работах (решении задач) по программированию студенту необходимо сделать математическое описание, написать алгоритм задачи любым из возможных способов, написать код программы.

Общие требования к форме отчетности по работам и заданиям

Отчеты-проекты по практическим работам оформляются в соответствии с требованиями методических указаний по выполнению каждой работы и требованиями стандарта СТО ИрГТУ 005 - 2007.

Отчет должен содержать титульный лист, цель работы, теоретическую часть, описание и схему лабораторной установки, порядок работы, таблицу измерений и расчетов, выводы по работе, ответы на контрольные вопросы, список литературы. Отчет должен быть оформлен на бумажном машинном носителе, формат А4.

Общие рекомендации по выполнению самостоятельной работы

Для успешного усвоения пройденного материала и активного участия в аудиторной работе студент должен готовиться к предстоящим занятиям, заранее знать тему занятия и повторить пройденный материал просмотрев этот раздел в рекомендованной литературе. Студент должен выполнить объем самостоятельной работы, предусмотренный рабочим учебным планом, максимально используя возможности индивидуального творческого и научного потенциала для освоения образовательной программы в целом.

При изучении теоретического материала в процессе самостоятельной подготовки для закрепления и расширения теоретических знаний, а также подготовки к проверке знаний необходимо следующее;

- изучение теоретического материала с помощью основной и дополнительной литературы и информационных ресурсов;
- изучение теоретической части при подготовке к проведению и защите практических работ.

6 Фонд оценочных средств для контроля текущей успеваемости и проведения промежуточной аттестации по дисциплине

6.1 Оценочные средства для проведения текущего контроля

6.1.1 семестр 3 | Отчет

Описание процедуры.

В каждом отчете о выполненном задании студенту необходимо привести математическое описание, блок-схему алгоритма, исходный код, итоговый результат работы в видео снимка экрана. Отчет выполняется на бумаге или в электронном виде в pdf формате.

Критерии оценивания.

Задание выполнено корректно, отчет был защищен

6.1.2 семестр 3 | Контрольная работа

Описание процедуры.

Задания для выполнения контрольной работы

В рамках выполнения письменной контрольной работы студент должен ответить на 1 контрольный вопрос и решить 1 задачу.

Контрольные вопросы в вариантах охватывают различные разделы курса. Студент отвечает на вопросы, соответствующие его варианту. Номер варианта указывается преподавателем.

Вопросы:

1. Каковы составляющие понятия «Программирование»?
2. Что входит в понятие «программный продукт» ?
3. Что такое «язык программирования» ?
4. Основные этапы разработки пользовательских программ, их содержание?
5. Модульное программирование, основная цель, понятие модуля, взаимодействие модулей?
6. Структурное программирование, основные положения?
7. Описание модуля языка , его части, объекты и их назначение?
8. Понятие подпрограммы, разновидности подпрограмм?
9. Простые типы данных, средства их задания, внешнее и внутреннее представление их значений, операции?
10. Понятие константы, типы, описание, использование?
11. Понятие переменной, типы, описание, использование?
12. Стандартные функции?
13. Операции ввода данных с клавиатуры и вывода на экран?
14. Реализация ветвящихся алгоритмов условными операторами?
15. Реализация повторяющихся алгоритмов операторами цикла с предусловием, постусловием и оператором цикла с параметром.
16. Понятие метки и оператора перехода?
17. Как создать и обрабатывать простейшее меню в текстовом режиме?
18. Использование случайных чисел в качестве входных данных при тестировании программ?
19. Понятие подпрограммы, назначение параметризация, механизм вызова подпрограмм?
20. Описание функции, формальные параметры, тип функции?
21. Описание процедуры, формальные параметры, вызов процедуры?
22. Формальный параметр вида параметр-значение, параметр-переменная, их возможные типы, соответствие фактическим параметрам?
23. Рекурсивные подпрограммы?
24. Файл, файловая переменная, последовательный и произвольный доступ?
25. Категории файлов и их свойства?
26. Стандартные процедуры и функции для всех файлов?
27. Стандартные процедуры и функции для текстовых файлов?
28. Стандартные процедуры и функции для типизированных файлов?
29. Внешние устройства в качестве файлов?
30. Идея объектно-ориентированного программирования?
31. Понятие объекта, иерархия объектов?
32. Сто такое сложные структуры данных?

Перечень контрольных заданий:

Задание 1

1. Дан массив X вещественных чисел. Элементы массива расположить в обратном порядке, найти наибольший и наименьший элементы массива.
 2. Понятие цикла. Цикл с параметром.
- Задание 2
1. Дан массив X вещественных чисел. Элементы массива циклически сдвинуть на одну позицию влево: $x_n = x_1$, $x_k = x_{k+1}$ при $k=1, 2, \dots, n-1$. Найти наибольший и наименьший элементы массива.
 2. Понятие цикла. Цикл с постусловием.
- Задание 3
1. Дан массив X . Преобразовать массив по следующему правилу: все отрицательные элементы массива X перенести в его начало, а все остальные в его конец, сохраняя исходное взаимное расположение.
 2. Понятие цикла. Цикл с предусловием
- Задание 4
1. Даны две последовательности по 30 чисел в каждой. Найти наименьшее среди тех чисел первой последовательности, которые не входят во вторую последовательность (считая, что хотя бы одно такое число есть).
 2. Понятие массива, описание массива.
- Задание 5
1. Дана последовательность из 20 целых чисел. Определить количество инверсий в этой последовательности (т. е. таких пар элементов, в которых большее число находится слева от меньшего: $x_i > x_j$ при $i < j$).
 2. Полное имя массива, номер элемента массива.
- Задание 6
1. Даны действительные числа $a_1, \dots, a_{20}$. Получить числа $b_1, \dots, b_{20}$, где b_i - среднее арифметическое всех членов последовательности $a_1, \dots, a_{20}$, кроме a_i ($i=1, 2, \dots, 20$).
 2. Условный оператор.
- Задание 7
1. Дано число a . Определить первый отрицательный член последовательности $x_1, x_2, x_3, \dots$, где $x_1 = a$, $x_n = \lg(x_{n-1})$.
 2. Одномерные массивы. Ввод данных в массив.
- Задание 8
1. Дано 50 чисел. Определить, сколько среди них отличных от последнего числа.
 2. Получение суммы и произведения элементов массива.
- Задание 9
1. Дано 50 чисел. Напечатать сначала все отрицательные из них, а затем все остальные.
 2. Переменная с индексом, размерность массива.
- Задание 10
1. Дано `const = 100; var x : array[1..n] of real;`
Элементы массива расположить в обратном порядке.
 2. Оператор перехода.
- Задание 11
1. Дано `const = 100; var x : array[1..n] of real;`
Элементы массива циклически сдвинуть на одну позицию влево: $x_n = x_1$, $x_k = x_{k+1}$.
 2. Оператор варианта.
- Задание 12
1. Переменной t присвоить значение `true`, если элементы массива X упорядочены строго по возрастанию, и значение `false` иначе.
 2. Сложные циклы.
- Задание 13
1. Переменной K присвоить либо номер первого вхождения U в массив X , либо число

$n+1$, если Y не входит в X .

X - массив n целых чисел.

2. Основные виды программных продуктов

Задание 14

1. Дана последовательность из 50 целых чисел. Найти сумму чисел этой последовательности , расположенных между максимальным и минимальным числами (в сумму включить и оба этих числа).

2. Стандартные функции. Выражения

Задание 15

1. Найти сумму $1! + 2! + 3! + \dots + 100!$.

2. Циклы. Параметр цикла. Тело цикла. Графическое изображение в блок-схеме

Задание 16

1. Найти сумму всех четных элементов массива , стоящих на четных местах, т.е. имеющих четные номера.

2. Логические переменные, логические операции

Задание 17

1. Найти номера всех отрицательных элементов массива, если таких нет, то сообщить об этом.

2. Что такое математическая модель задачи?

Задание 18

1. Создать массив по следующему правилу: если элемент четный , то прибавить к нему первый, если нечетный, то прибавить к нему последний. Первый и последний элементы не изменять.

2. Алфавит, константы, переменные языка C++

Задание 19

1. Изменить знак у максимального по модулю элемента массива.

2. Структурное программирование

Задание 20

1. Заменить все четные элементы на их квадраты, а нечетные удвоить.

2. Стандартные функции. Выражения

Задание 21

1. Даны два целочисленных массива, состоящие из одинакового числа элементов.

Получить третий массив той же размерности, каждый элемент которого равен большему из соответствующих элементов данных массивов.

2. Ввод и вывод данных

Задание 22

1. Дана последовательность из n целых чисел. Какое число встречается в ней раньше – положительное или отрицательное. Если все числа равны 0 , сообщить об этом.

2. Операции сравнения. Логические операции

Задание 23

1. Дана последовательность из n целых чисел. Каких чисел больше – делящихся на 3 или на 7. Если ни одно из чисел не делится ни на 3, ни на 7 , сообщить об этом.

2. Оператор выбора, оператор перехода

Задание 24

1. Удалить последний четный элемент

2. Массивы. Описание массива

Критерии оценивания.

Правильно выполненное задание и корректный ответ на вопрос - зачтено.

6.2 Оценочные средства для проведения промежуточной аттестации

6.2.1 Критерии и средства (методы) оценивания индикаторов достижения компетенции в рамках промежуточной аттестации

Индикатор достижения компетенции	Критерии оценивания	Средства (методы) оценивания промежуточной аттестации
ОПК ОС-8.1	Знает основные виды и этапы проектирования программных продуктов. Умеет четко и ясно формулировать задачу. Может разработать алгоритм для решения задачи, в виде четких и последовательных шагов. Понимает принципы структурного и модульного программирования, и умеет применять их на практике. Знает методы и средства объектно-ориентированного программирования, умеет применять ООП-подход при разработке программ. Владеет синтаксисом и семантикой языка программирования C++. 1. Может создать и отладить программу.	Выполнение контрольного индивидуального задания; электронное тестирование

6.2.2 Типовые оценочные средства промежуточной аттестации

6.2.2.1 Семестр 3, Типовые оценочные средства для проведения экзамена по дисциплине

6.2.2.1.1 Описание процедуры

Вопросы к экзамену

1. Программирование. Этапы проектирования программных продуктов.
2. Дан массив, состоящий из 15 элементов. Из положительных элементов массива извлечь квадратный корень, а отрицательные возвести в 5-ю степень.
3. Описание переменных в СИ-программе.
4. Этапы подготовки и решения задач на ЭВМ? Стили и методы программирования.
5. Вычислить значение функции: $y = \begin{cases} 1^x, & x < 6 \\ \sqrt{x+2}, & x = 6 \\ \ln x, & x > 6 \end{cases}$
6. Понятие цикла. 3 типа циклов.
7. Алфавит, константы, переменные языка C++.
8. Вычислить произведение каждой строки матрицы $A(m, n)$.
9. Что такое математическая модель задачи?
10. Константы. Типы констант и правила их записи в C++.
11. Вычислить: $y = (\ln|x + (a+b)/\sin x| - 4,7 \cdot 10^{-8}) / (\sqrt{5x^3 + 12} + \arctg x) - 1^x$
12. Модульное программирование
13. Переменные. Типы переменных. Правила их задания в C++.

14. Дан массив , состоящий из 10 элементов. Каждый второй элемент массива возвести в квадрат, вычислить их сумму и произведение.
15. Структурное программирование
16. Стандартные функции. Операторы. Выражения.
17. Вычислить: $y_i = \{ \begin{cases} \lfloor \sin \rfloor^2 (y_{(i-1)} \times z + l^y)^3, & y_{(i-1)} \geq z @ l^{\wedge}(z/y), \\ y_{(i-1)} < z \end{cases} \}$, где $i=0..n$
18. Модульное программирование
19. Ввод исходных данных в ЭВМ.
20. Сформировать массив z, вычислив его элементы по правилу: $Z_i = (\cos(i))/i^2$, где $i=1,2,...,20$
21. Управляющие структуры.
22. Алгоритм. Описание алгоритма.
23. Вычислить: $y_i = \{ \begin{cases} \lfloor \cos \rfloor^2 \lfloor x_i^3 \rfloor + 17, & 3, x_i \leq c @ 1/\sqrt[3]{(x_i)}, \\ x_i > c \end{cases} \}$, где $I=1,2,...,15$
24. Сложные циклы
25. Вывод данных. Форматный вывод
26. Дан массив , состоящий из n элементов. Подсчитать количество элементов, для которых выполняется условие $-3 \leq X_i \leq 8$
27. Массивы. Описание массива
28. Понятие оператора. Условный оператор .
29. Составить программу : все отрицательные элементы массива X перенести в его начало, а все остальные - в конец, сохраняя исходное взаимное расположение (использовать вспомогательный массив).
30. Схема подготовки исполняемой программы.
31. Простые типы данных.
32. Вычислить сумму членов ряда $X^2 + X^4 + \dots + X^{24}$
33. Оператор выбора, операторы перехода.
34. Понятие оператора. Условный оператор.
35. Вычислить: $y_i = \{ \begin{cases} 1/(a-p_i) \ln a, & a \geq 2 @ 2a - \sin \lfloor b_i \rfloor, \\ a < 2 \end{cases} \}$
36. Операции сравнения. Логические операции.
37. Разветвляющиеся вычислительные процессы. Конструкции если-то-иначе, если-то
38. Дан массив A, состоящий из 12 элементов. Подсчитать сколько из этих элементов принадлежат интервалу [1,2], найти их сумму и произведение
39. Ввод и вывод данных
40. Выполнение математических операций. Бинарная и унарная арифметика.
41. Даны a, b, c - стороны треугольника. Найти медианы треугольника.
42. Типы данных.
43. Циклы. Параметр цикла. Тело цикла. Графическое изображение в блок-схеме.
44. Вычислить: $y_i = \{ \begin{cases} \sqrt{(x_i^3)}, & x_i < b @ 1+b, \\ x_i = b @ \lfloor \sin \rfloor^2 \lfloor x_i \rfloor, \\ x_i > b \end{cases} \}$
45. Функция в C++. Определение и описание функции.
46. Массивы. Типы массивов. Описание массива.
47. Вычислить $(1+\sin 0,1)(1+\sin 0,2) \dots (1+\sin 0,10)$.
48. Логические переменные, логические операции
49. Функция в C++. Функция main.
50. Вычислить $(1+\sin 0,1)(1+\sin 0,2) \dots (1+\sin 0,10)$.
51. Логические переменные, логические операции
52. Функция в C++. Оператор return.
53. Дан массив A, состоящий из 12 элементов. Подсчитать сколько из этих элементов принадлежат интервалу [1,2], найти их сумму и произведение
54. Символьные переменные.
55. Функция в C++. Прототип функции. Правила работы с прототипами.

56. Вычислить: $y_i = \begin{cases} \sqrt{x_i^3}, & x_i < b \\ 1+b, & x_i = b \\ \sin^2 x_i, & x_i > b \end{cases}$
57. Алгоритм. Свойства алгоритма. Описание алгоритма.
58. Функция в C++. Обращение к функции.
59. Вычислить: $y_i = \begin{cases} \sqrt{x_i^3}, & x_i < b \\ 1+b, & x_i = b \\ \sin^2 x_i, & x_i > b \end{cases}$
60. Математическая модель задачи.

6.2.2.1.2 Критерии оценивания

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Знает основные виды и этапы проектирования программных продуктов. Умеет четко и ясно формулировать задачу. Может разработать алгоритм для решения задачи, в виде четких и последовательных шагов. Понимает принципы структурного и модульного программирования, и умеет применять их на практике. Знает методы и средства объектно-ориентированного программирования, умеет применять ООП-подход при разработке программ. Владеет синтаксисом и семантикой языка программирования C++. 1. Может создать и отладить программу.	Знает основные виды и этапы проектирования программных продуктов. Умеет четко и ясно формулировать задачу. Может разработать алгоритм для решения задачи. Понимает принципы структурного и модульного программирования, и умеет применять некоторые из них на практике. Знает методы и средства объектно-ориентированного программирования, не всегда может применить ООП при разработке программ. Владеет синтаксисом и семантикой языка программирования C++. 1. Может создать и отладить программу.	Знает базовые этапы проектирования программных продуктов. Формулирует задачу с некоторыми оговорками. Разрабатывает алгоритм для решения типовых задач. Понимает общие принципы структурного и модульного программирования. Знает методы и средства объектно-ориентированного программирования. Имеет представление о синтаксисе и семантике языка программирования C++. 1. Может создать и отладить программу.	Нет понятий о этапах проектирования программных продуктов. Формулировка задачи вызывает сложности. Разработка алгоритмов вызывает затруднения. Нет понятия структурного и модульного программирования, объектно-ориентированного программирования. Имеет частичное представление о синтаксисе и семантике языка программирования C++. 1. Не может создать и отладить программу.

7 Основная учебная литература

1. Колдаев В. Д. Основы алгоритмизации и программирования [Электронный ресурс] : учебное пособие / В. Д. Колдаев ; под ред. проф. Л. Г. Гагариной, 2022. - 414.

2. Аршинский В. Л. Программирование на языке высокого уровня. Основы алгоритмизации и программирования : учебное пособие / В. Л. Аршинский, И. А. Серышева, 2021. - 122.

3. Основы алгоритмизации и программирования [Электронный ресурс] : методические указания по выполнению практических работ для студентов специальности 09.02.07 Информационные системы и программирование / ИРНИТУ, 2022. - 166.

8 Дополнительная учебная литература и справочная

1. Трофимов В. В. Основы алгоритмизации и программирования : учебник для студентов среднего специального образования / В. В. Трофимов, Т. А. Павловская, 2021. - 137.

2. Жданова Т. А. Основы алгоритмизации и программирования : учебное пособие / Т. А. Жданова, Ю. С. Бузыкова, 2012. - 55.

3. Парфилова Н. И. Программирование. Основы алгоритмизации и программирования : учебник по направлению подготовки "Информатика и вычислительная техника" / Н. И. Парфилова, А. Н. Пылькин, Б. Г. Трусков, 2014. - 239.

9 Ресурсы сети Интернет

1. <http://library.istu.edu/>

2. <https://e.lanbook.com/>

10 Профессиональные базы данных

1. <http://new.fips.ru/>

2. <http://www1.fips.ru/>

11 Перечень информационных технологий, лицензионных и свободно распространяемых специализированных программных средств, информационных справочных систем

1. Лицензионное программное обеспечение Системное программное обеспечение

2. Лицензионное программное обеспечение Пакет прикладных офисных программ

3. Лицензионное программное обеспечение Интернет-браузер

4. Microsoft Visual ++

12 Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения лекционных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.

2. Учебная аудитория для проведения лабораторных/практических (семинарских) занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска.

Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.