



УТВЕРЖДЕНО:
Ученым советом Института
Сервисных технологий
Протокол № 7 от «15» января 2026г.

РАБОЧАЯ ПРОГРАММА
ДИСЦИПЛИНЫ (СПО)
ОП.06 ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ
основной профессиональной образовательной программы среднего
профессионального образования – программы подготовки специалистов среднего
звена
по специальности: *09.02.11 Разработка и управление программным
обеспечением*
Квалификация: *Программист*
год начала подготовки: *2026*

Разработчики:

| должность | ученая степень и звание, ФИО |
|----------------------|------------------------------|
| <i>преподаватель</i> | <i>Коваленко М.В.</i> |

Рабочая программа согласована и одобрена руководителем ППСЗ:

| должность | ученая степень и звание, ФИО |
|----------------------|------------------------------|
| <i>преподаватель</i> | <i>Границына М.С</i> |



СОДЕРЖАНИЕ

| | | |
|----------|--|-----------|
| 1 | Общая характеристика рабочей программы дисциплины | 3 |
| 2 | Структура и содержание учебной дисциплины | 6 |
| 3 | Методические указания по проведению практических занятий/лабораторных работ/семинаров, <u>занятий в форме практической подготовки (при наличии)</u>, и самостоятельной работе | 14 |
| 4 | Фонд оценочных средств дисциплины | 19 |
| 5 | Условия реализации программы дисциплины | 36 |
| 6 | Информационное обеспечение реализации программы | 36 |



1. Общая характеристика рабочей программы дисциплины
«ОП.06 Основы алгоритмизации и программирования»
(наименование дисциплины)

1.1 Место дисциплины в структуре основной образовательной программы:

Учебная дисциплина ОП.06 «Основы алгоритмизации и программирования» является обязательной частью цикла общепрофессиональных дисциплин примерной основной образовательной программы в соответствии с ФГОС по специальности 09.02.11 Разработка и управление программным обеспечением.

Особое значение дисциплина имеет при формировании и развитии ОК: ОК 01, ОК 02, ОК 03, ОК 04, ОК 09, ПК2.1 – ПК2.5.

Программа учебной дисциплины может быть использована в дополнительном профессиональном образовании в рамках реализации программ подготовки и переподготовки кадров в учреждениях СПО.

Учебная дисциплина ОП.06 «Основы алгоритмизации и программирования» относится к циклу общепрофессиональных дисциплин основной профессиональной образовательной программы.

1.2 Цель и планируемые результаты освоения дисциплины:

В результате освоения дисциплины обучающийся должен уметь:

- Разрабатывать алгоритмы для конкретных задач.
- Использовать программы для графического отображения алгоритмов.
- Определять сложность работы алгоритмов.
- Работать в среде программирования.
- Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.
- Оформлять код программы в соответствии со стандартом кодирования.
- Выполнять проверку, отладку кода программы.

В результате освоения дисциплины обучающийся должен знать:

- Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.
- Эволюцию языков программирования, их классификацию, понятие системы программирования.
- Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.
- Подпрограммы, составление библиотек подпрограмм
- Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения

В рамках программы учебной дисциплины обучающимися осваиваются умения и знания:

| Код ПК, ОК | Умения | Знания |
|---------------|--------|--------|
|---------------|--------|--------|



| | | |
|--|--|---|
| ОК 01 ОК 02 ОК 03 ОК 04 ОК 09 ПК2.1- ПК2.5 | <ul style="list-style-type: none">– Разрабатывать алгоритмы для конкретных задач.– Использовать программы для графического отображения алгоритмов.– Определять сложность работы алгоритмов.– Работать в среде программирования.– Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования.– Оформлять код программы в соответствии со стандартом кодирования.– Выполнять проверку, отладку кода программы. | <ul style="list-style-type: none">– Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции.– Эволюцию языков программирования, их классификацию, понятие системы программирования.– Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти.– Подпрограммы, составление библиотек подпрограмм– Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения |
|--|--|---|

2. Структура и содержание учебной дисциплины

2.1. Объем учебной дисциплины и виды учебной работы

| Вид учебной работы | Объем в часах |
|--|---------------|
| Объем образовательной программы учебной дисциплины | 142 |
| <i>в т.ч. в форме практической подготовки (если предусмотрено)</i> | |
| в т. ч.: | |
| теоретическое обучение | 52 |
| лабораторные работы (если предусмотрено) | |
| практические занятия (если предусмотрено) | 70 |
| курсовая работа (проект) (если предусмотрено для специальностей) | |
| контрольная работа (если предусмотрено) | |
| консультации | 2 |
| Самостоятельная работа ¹ | 6 |
| Промежуточная аттестация (экзамен 4 семестр) | 12 |

¹ Самостоятельная работа в рамках образовательной программы планируется образовательной организацией в соответствии с требованиями ФГОС СПО в пределах объема учебной дисциплины в количестве часов, необходимом для выполнения заданий самостоятельной работы обучающихся, предусмотренных тематическим планом и содержанием учебной дисциплины.



2.2. Тематический план и содержание учебной дисциплины Основы алгоритмизации и программирования

наименование

| Наименование разделов и тем | Примерное содержание учебного материала, практических и лабораторных занятий, курсовой проект (работа) | Объем в часах | Коды компетенций, формированию которых способствует элемент программы |
|---|---|---------------|---|
| Раздел 1. Введение в программирование | | | |
| Тема 1.1. Основы алгоритмизации, языки и системы программирования Наименование Основные элементы языка. Типы данных. Основы структурного программирования. | Содержание | | ОК 01- ОК 09, ПК 2.1- ПК 2.5 |
| | Алгоритм и его свойства. Виды алгоритмов. Способы записи алгоритмов. Сложность алгоритмов Эволюция и классификация языков программирования. Среда программирования. Компиляторы и интерпретаторы. Жизненный цикл программы. Программа. Программный продукт и его характеристики.) | 10 | |
| | Основные элементы языка. Типы данных. Простые типы данных. Производные типы данных. Структурированные типы данных. Операторы языка. Ввод/вывод данных. Обработка исключений. Операторы разветвляющихся программ. | 10 | |
| | Циклические программы. Одномерные массивы. Двумерные массивы. Работа с массивами. Сортировка массивов. | 6 | |
| | Строки. Коллекции. Файлы. Доступ к файлам. Виды файлов. Считывание и запись в файл. | 4 | |
| | В том числе практических и лабораторных занятий | | |
| | Линейные программы | 2 | |
| | Составление программ разветвляющейся структуры | 2 | |
| | Циклические программы | 2 | |
| | Одномерные массивы | 2 | |
| | Двумерные массивы. | 2 | |
| | Обработка массивов | 2 | |
| Сортировка массивов | 2 | | |



| | | | |
|--|--|----------|---|
| | Символы и строки. Обработка строк. | 4 | |
| | Использование коллекций | 2 | |
| | Работа с файлами | 2 | |
| | Работа с файлами | 2 | |
| | Работа с каталогами и файлами | 2 | |
| | В том числе самостоятельная работа обучающихся <i>Необходимость и тематика определяются образовательной организацией</i> | 2 | |
| Раздел 2. Технологии программирования | | | ОК 01- ОК 09, ПК 2.1- ПК 2.5 |
| Тема 2.1. Модульное программирование. | Содержание | | |
| | Модульное программирование Локальные и глобальные переменные Подпрограммы. Модификаторы. Передача данных в подпрограммы. Рекурсия. Разработка рекурсивных подпрограмм. | 8 | |
| | В том числе практических и лабораторных занятий | | |
| | Использование подпрограмм. | 2 | |
| | Использование подпрограмм. | 2 | |
| | Рекурсия | 2 | |
| | Создание модулей | 4 | |
| | В том числе самостоятельная работа обучающихся <i>Необходимость и тематика определяются образовательной организацией</i> | 2 | |
| Тема 2.2. Основные принципы объектно- ориентированного программирования | Содержание | | |
| | Базовые понятия ООП: объект, его свойства и методы, класс, интерфейс. Основные принципы ООП: инкапсуляция, наследование, полиморфизм. | 6 | |
| | В том числе практических и лабораторных занятий | | |
| | Работа с классами. Создание конструкторов. | 4 | |
| | Применение свойств | 2 | |
| | Наследование | 2 | |
| Полиморфизм | 2 | | |



ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТУРИЗМА И СЕРВИСА»

СМК
РГУТИС

Лист 7

| | | | |
|--|---|-----------|---|
| | В том числе самостоятельная работа обучающихся <i>Необходимость и тематика определяются образовательной организацией</i> | 1 | |
| Раздел 3 Разработка приложений | | | ОК 01- ОК 09, ПК 2.1- ПК 2.5 |
| Тема 3.1. Этапы разработки приложений | Содержание | | |
| | Визуально-событийно управляемое программирование. Разработка приложения. Проектирование объектно-ориентированного приложения. Создание интерфейса пользователя. Тестирование, отладка приложения. Оптимизация программы | 8 | |
| | В том числе практических и лабораторных занятий | | |
| | Создание проекта с использованием компонентов для работы с текстом | 4 | |
| | Создание проекта с использованием кнопочных компонентов | 4 | |
| | Создание проекта с использованием переключателей | 2 | |
| | Создание проекта с использованием компонентов для отображения таблиц | 4 | |
| | Создание проекта с использованием компонентов для отображения дат и времени | 4 | |
| | Разработка интерфейса приложения | 2 | |
| | Разработка интерфейса приложения | 2 | |
| | Тестирование приложения | 2 | |
| В том числе самостоятельная работа обучающихся <i>Необходимость и тематика определяются образовательной организацией</i> | 1 | | |
| Промежуточная аттестация | | 12 | |
| Всего 142 часа | | | |



3. Методические указания по проведению практических занятий/лабораторных работ/семинаров, занятий в форме практической подготовки (при наличии) и самостоятельной работе

Практические занятия проводятся в компьютерном классе в формах:

- 1) индивидуальная самостоятельная работа по заданию;
- 2) выполнение проверочных контрольных работ;
- 3) обучающий тренинг.

3.1. Тематика и содержание практических занятий

Тематика практических занятий соответствует рабочей программе дисциплины.

Практическое занятие

Тема: Основы алгоритмизации, языки и системы программирования
Наименование: Основные элементы языка. Типы данных. Основы структурного программирования.

Содержание:

| |
|--|
| Линейные программы |
| Составление программ разветвляющейся структуры |
| Циклические программы |
| Одномерные массивы |
| Двумерные массивы. |
| Обработка массивов |
| Сортировка массивов |
| Символы и строки. Обработка строк. |
| Использование коллекций |
| Работа с файлами |
| Работа с каталогами и файлами |

Практическое занятие

Тема: **Модульное программирование.**

Содержание:

| |
|----------------------------|
| Использование подпрограмм. |
| Использование подпрограмм. |
| Рекурсия |
| Создание модулей |

Практическое занятие

Тема: **Основные принципы объектно-ориентированного программирования**

Содержание:

| |
|--|
| Работа с классами. Создание конструкторов. |
| Применение свойств |
| Наследование |
| Полиморфизм |



Практическое занятие

Тема: **Этапы разработки приложений**

Содержание:

| |
|---|
| Создание проекта с использованием компонентов для работы с текстом |
| Создание проекта с использованием кнопочных компонентов |
| Создание проекта с использованием переключателей |
| Создание проекта с использованием компонентов для отображения таблиц |
| Создание проекта с использованием компонентов для отображения дат и времени |
| Разработка интерфейса приложения |
| Тестирование приложения |

3.2. Тематика и содержание самостоятельной работы

На самостоятельную работу студентов выделяется 6 академических часов в рамках часов на дисциплину Основы алгоритмизации и программирования.

Тема: Проработка конспекта лекций по теме, изучение дополнительной литературы и источников сети интернет.

Содержание: Ознакомление с рекомендованными источниками по дисциплине.

4. Фонд оценочных средств дисциплины

4.1. Результаты освоения учебной дисциплины, подлежащие проверке

Формы промежуточной аттестации по семестрам:

| № семестра | Форма контроля |
|------------|--------------------------|
| 3 | Дифференцированный зачет |
| 4 | экзамен |

В результате промежуточной аттестации по учебной дисциплине осуществляется комплексная проверка следующих умений и знаний, а также формирования компетенций:

| Результаты обучения: умения, знания | Показатели оценки результата | Форма контроля и оценивания |
|---|---|--|
| Уметь: | | |
| Разрабатывать алгоритмы для конкретных задач. | Умеет разрабатывать алгоритмы для конкретных задач. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, оценка выполнения контрольных работ, устный опрос, тестирование. <i>Для промежуточной аттестации:</i> экзамен |
| Использовать программы для | Умеет использовать | <i>Для текущего контроля:</i> |



| | | |
|---|---|--|
| графического отображения алгоритмов. | программы для графического отображения алгоритмов. | оценка работы на практических занятиях, оценка выполнения контрольных работ, устный опрос, тестирование. <i>Для промежуточной аттестации:</i> экзамен |
| Определять сложность работы алгоритмов. | Умеет определять сложность работы алгоритмов. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, оценка выполнения контрольных работ, устный опрос, тестирование. <i>Для промежуточной аттестации:</i> экзамен |
| Работать в среде программирования | Умеет работать в среде программирования | <i>Для текущего контроля:</i> оценка работы на практических занятиях, оценка выполнения контрольных работ, устный опрос, тестирование. <i>Для промежуточной аттестации:</i> экзамен |
| Реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. | Умеет реализовывать построенные алгоритмы в виде программ на конкретном языке программирования. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, оценка выполнения контрольных работ, устный опрос, тестирование. <i>Для промежуточной аттестации:</i> экзамен |
| Оформлять код программы в соответствии со стандартом кодирования. | Умеет оформлять код программы в соответствии со стандартом кодирования. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, оценка выполнения контрольных работ, устный опрос, тестирование. <i>Для промежуточной аттестации:</i> экзамен |
| Выполнять проверку, отладку кода программы. | Умеет выполнять проверку кода программы | <i>Для текущего контроля:</i> оценка работы на практических занятиях, оценка выполнения контрольных работ, устный опрос, тестирование. <i>Для промежуточной аттестации:</i> экзамен |
| Знать: | | |
| Понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные | Знает понятие алгоритмизации, свойства алгоритмов, общие принципы построения | <i>Для текущего контроля:</i> фронтальный опрос <i>Для промежуточной аттестации:</i> |



| | | |
|---|---|---|
| алгоритмические конструкции. | алгоритмов, основные алгоритмические конструкции. | Контрольная работа, экзамен |
| Эволюцию языков программирования, их классификацию, понятие системы программирования | Знает эволюцию языков программирования, их классификацию, понятие системы программирования | <i>Для текущего контроля:</i> фронтальный опрос <i>Для промежуточной аттестации:</i> Контрольная работа, экзамен |
| Основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. | Знает основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти. | <i>Для текущего контроля:</i> фронтальный опрос <i>Для промежуточной аттестации:</i> Контрольная работа, экзамен |
| Подпрограммы, составление библиотек подпрограмм | Знает подпрограммы, составление библиотек подпрограмм | <i>Для текущего контроля:</i> фронтальный опрос <i>Для промежуточной аттестации:</i> Контрольная работа, экзамен |
| Объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения | Знает объектно-ориентированную модель программирования, основные принципы объектно-ориентированного программирования на примере алгоритмического языка: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения | <i>Для текущего контроля:</i> фронтальный опрос <i>Для промежуточной аттестации:</i> Контрольная работа, экзамен |

Формируемые компетенции:

| Код формируемой компетенции | Наименование компетенции | Формы и методы контроля и оценки результатов обучения |
|-----------------------------|---|---|
| ОК 1. | Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ОК 2. | . Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> |



| | профессиональной деятельности | экзамен |
|---------|---|---|
| ОК 3. | Планировать и реализовывать собственное профессиональное и личностное развитие, предпринимательскую деятельность в профессиональной сфере, использовать знания по правовой и финансовой грамотности в различных жизненных ситуациях | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ОК 4 | Эффективно взаимодействовать и работать в коллективе и команде | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ОК 09. | Пользоваться профессиональной документацией на государственном и иностранном языке. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ПК 2.1. | Проектировать модули программного обеспечения. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ПК 2.2. | Разрабатывать модули программного обеспечения. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ПК 2.3. | Выполнять интеграцию модулей и компонентов программного обеспечения | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ПК 2.4. | Выполнять тестирование и отладку программного обеспечения. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> экзамен |
| ПК 2.5. | Осуществлять документирование программных модулей программного обеспечения. | <i>Для текущего контроля:</i> оценка работы на практических занятиях, устный опрос. <i>Для промежуточной аттестации:</i> |



| | | |
|--|--|------------------------|
| | | аттестации: экзамен |
|--|--|------------------------|

4.2. Методика применения контрольно-измерительных материалов

Формы контроля

Текущий контроль осуществляется в течение семестра в виде оценки результатов практических занятий, оценки выполнения контрольных работ, тестов, оценки устных опросов.

Промежуточная аттестация осуществляется в виде дифференцированного зачета в 3 семестре и экзамена в 4 семестре

4.3. Контрольно-измерительные материалы включают:

4.3.1. Типовые задания для оценки знаний и умений текущего контроля

Контроль и оценка результатов освоения темы осуществляется преподавателем в процессе выполнения обучающимися индивидуальных заданий в виде внешнего контроля. Формы контроля: проверка практического задания, тестирование.

Типовые задания для текущего контроля оценки знаний и умений

Примерные тестовые задания:

- 1. Что такое алгоритм?**
 - а) Язык программирования
 - б) Конечная последовательность шагов для решения задачи
 - в) Математическая формула
 - г) Инструкция по сборке мебели
- 2. Какое свойство алгоритма означает, что он должен приводить к результату за конечное число шагов?**
 - а) Дискретность
 - б) Массовость
 - в) Результативность
 - г) Дискретность
- 3. Свойство алгоритма, означающее, что каждое действие должно быть четко определено:**
 - а) Массовость
 - б) Дискретность
 - в) Определенность (детерминированность)
 - г) Понятность
- 4. Какая алгоритмическая структура используется для выполнения действий в зависимости от условия?**
 - а) Линейная
 - б) Ветвление
 - в) Цикл
 - г) Последовательность
- 5. Алгоритм, в котором команды выполняются последовательно одна за другой, называется:**
 - а) Ветвящимся
 - б) Циклическим
 - в) Линейным
 - г) Вспомогательным
- 6. Графическое представление алгоритма — это:**
 - а) Программа
 - б) Блок-схема



- c) Псевдокод
d) Текст
7. **Что означает свойство «массовость» алгоритма?**
a) Алгоритм применяется к одной задаче
b) Алгоритм применим к классу однотипных задач
c) Алгоритм содержит много команд
d) Алгоритм написан для множества исполнителей
8. **Какая геометрическая фигура в блок-схеме обозначает начало/конец алгоритма?**
a) Прямоугольник
b) Ромб
c) Овал
d) Параллелограмм
9. **Ромб в блок-схемах используется для обозначения:**
a) Действия
b) Ввода данных
c) Условия (ветвления)
d) Вывода результатов
10. **Алгоритм, включающий в себя подпорядки, называется:**
a) Линейным
b) Вспомогательным
c) Циклическим
d) Структурным (процедурным) [[1](#), [2](#), [3](#), [4](#), [5](#)]
- Блок 2: Основы программирования и типы данных (11-20)**
11. **Что такое язык программирования?**
a) Язык общения программистов
b) Формальный язык для записи алгоритмов
c) Среда разработки
d) Машинный код
12. **Как называется процесс перевода программы с языка высокого уровня на язык машинных кодов?**
a) Отладка
b) Компиляция
c) Редактирование
d) Проектирование
13. **Какой тип данных используется для хранения целых чисел?**
a) real
b) char
c) integer
d) boolean
14. **Какой тип данных используется для хранения символов?**
a) string
b) char
c) integer
d) float
15. **Логический тип данных (boolean) может принимать значения:**
a) Да/Нет
b) 0/1



- c) True/False
d) Все вышеперечисленное
16. **Что такое переменная?**
a) Постоянная величина
b) Именованная область памяти для хранения данных
c) Функция
d) Тип данных
17. **Какой оператор обычно используется для присваивания значения переменной?**
a) =
b) :=
c) ==
d) Верно а) и б) в зависимости от языка
18. **Какой из перечисленных языков является языком высокого уровня?**
a) Ассемблер
b) Машинный код
c) Python
d) Двоичный код
19. **Какой тип данных лучше использовать для хранения веса товара?**
a) integer
b) char
c) real (или float)
d) boolean
20. **Константа — это:**
a) Величина, меняющаяся в ходе программы
b) Величина, не меняющаяся в ходе программы
c) Тип данных
d) Оператор [1, 2, 3]
- Блок 3: Операции и линейные алгоритмы (21-30)**
21. **Результат выражения $\backslash(10 \backslash \text{div } 3)$ (целочисленное деление):**
a) 3.33
b) 3
c) 1
d) 0
22. **Результат операции $10 \bmod 3$ (остаток от деления):**
a) 3
b) 1
c) 0
d) 30
23. **Какая операция имеет наивысший приоритет?**
a) +
b) *
c) ()
d) /
24. **Чему равно $15 / 2$ (вещественное деление)?**
a) 7
b) 7.5
c) 1
d) 8



25. Каков результат логического выражения $(5 > 3)$ and $(2 > 4)$?
- True
 - False
 - 1
 - 0
26. Что делает оператор `input()` (в Python)?
- Выводит данные на экран
 - Считывает данные с клавиатуры
 - Очищает экран
 - Завершает программу
27. Какое значение примет переменная (X) после выполнения: $(X:=5)$; $(X:=X+10)$;
- 5
 - 10
 - 15
 - 0
28. Каков результат логического выражения $(5 > 3)$ or $(2 > 4)$?
- True
 - False
 - 1
 - 0
29. Операция `not` в логических выражениях означает:
- И
 - Или
 - Отрицание
 - Сравнение
30. Как правильно записать $(a = \frac{b}{c})$ в программировании?
- `a = b \ c`
 - `a = b / c`
 - `a := b / c`
 - `a = b : c` [1]

Блок 4: Условные операторы и циклы (31-40)

31. Какой оператор используется для организации ветвления?
- `for`
 - `if`
 - `while`
 - `repeat`
32. Что произойдет, если в цикле `while` условие всегда истинно?
- Цикл не выполнится
 - Программа зависнет (бесконечный цикл)
 - Цикл выполнится 1 раз
 - Ошибка компиляции
33. Какой цикл используется, когда количество итераций заранее неизвестно?
- `for`
 - `while` (ПОКА)
 - Цикл с параметром
 - Таблица
34. Какой цикл используется, когда количество итераций известно?
- `while`



- b) for (ДЛЯ)
c) if
d) switch
35. **Какая структура используется для многовариантного выбора?**
a) if-then-else
b) case (или switch)
c) while
d) for
36. **Сколько раз выполнится цикл: for i := 1 to 5 do?**
a) 4
b) 5
c) 6
d) 0
37. **Что такое «тело цикла»?**
a) Условие цикла
b) Операторы, повторяющиеся в цикле
c) Имя цикла
d) Начало цикла
38. **Какое значение будет выведено: if 5>3 then print('A') else print('B');?**
a) A
b) B
c) AB
d) Ничего
39. **Цикл с предусловием — это:**
a) for
b) while
c) do-while
d) repeat-until
40. **Цикл с постусловием — это:**
a) while
b) for
c) repeat-until (или do-while)
d) if [1, 2]

Блок 5: Массивы, функции и алгоритмы (41-50)

41. **Массив — это:**
a) Набор переменных разных типов
b) Упорядоченная совокупность однотипных элементов
c) Функция
d) Условный оператор
42. **Как обратиться к третьему элементу массива A?**
a) A[2]
b) A[3]
c) A(3)
d) A-3
43. **Индекс массива — это:**
a) Значение элемента
b) Порядковый номер элемента



- c) Тип элемента
d) Размер массива
44. **Какой алгоритм сортировки работает методом обмена соседних элементов?**
a) Сортировка выбором
b) Пузырьковая сортировка
c) Быстрая сортировка
d) Сортировка вставками
45. **Что такое функция в программировании?**
a) Имя программы
b) Подпрограмма, возвращающая значение
c) Тип данных
d) Цикл
46. **Какая структура данных используется для хранения двумерного массива?**
a) Линейная
b) Таблица (матрица)
c) Список
d) Дерево
47. **Для чего используется процедура?**
a) Для возврата значения
b) Для выполнения группы команд (без явного возврата значения)
c) Для описания констант
d) Для ввода данных
48. **Что такое рекурсия?**
a) Цикл
b) Вызов функции самой из себя
c) Ошибка
d) Сортировка
49. **Какая структура данных работает по принципу FIFO (первый пришел — первый ушел)?**
a) Стек
b) Очередь
c) Массив
d) Дерево
50. **Какая структура данных работает по принципу LIFO (последний пришел — первый ушел)?**
a) Очередь
b) Стек
c) Список
d) Массив

4.3.2. Типовые задания для оценки знаний и умений промежуточной аттестации Перечень теоретических вопросов для проведения промежуточной аттестации:

1. Что такое алгоритм? Дайте определение.
2. Назовите и поясните основные свойства алгоритма: дискретность, детерминированность, понятность, результативность, массовость.
3. Что означает свойство «конечность» (результативность) алгоритма?.
4. Какие существуют формы записи алгоритмов?.



5. Что такое блок-схема и какие основные элементы в ней используются?.
6. Как изображается структура «ветвление» (условие) в блок-схеме?
7. Как изображается структура «цикл» в блок-схеме?
8. Чем словесное описание алгоритма отличается от псевдокода? Что такое линейный алгоритм? Приведите пример.
9. Что такое алгоритмическая структура «ветвление» (полное и неполное)?.
10. Как работает условный оператор if-then-else?
11. Что такое выбор (оператор switch или case)? В каких случаях его лучше использовать?.
12. Что такое цикл? Какие типы циклов вы знаете?.
13. Что такое цикл с предусловием (while)? Когда тело цикла не выполнится ни разу?.
14. Что такое цикл с постусловием (do-while / repeat-until)? Сколько раз выполняется тело цикла как минимум?
15. Что такое цикл с параметром (for)? Какова его структура?.
16. Что такое «вложенный цикл»?
18. Что такое язык программирования? Чем отличаются языки высокого и низкого уровней?
19. Что такое переменная? Какие компоненты она имеет (имя, тип, значение, адрес)?.
20. Назовите базовые типы данных: целые, вещественные, символьные, логические.
21. Чем отличается тип данных int от float/double?
22. Что такое логический тип данных (boolean) и какие значения он принимает?
23. Что такое оператор присваивания? Приведите пример.
24. Как осуществляется ввод и вывод данных в языке программирования?
25. Что такое арифметическое выражение? Порядок выполнения операций.
26. Что такое целочисленное деление (div//) и остаток от деления (mod/%)?.
27. Что такое логическое выражение? Приведите примеры логических операций (AND, OR, NOT).
28. Что такое «область видимости» переменной (локальные и глобальные)?.
29. Что такое массив? Зачем он нужен?.
30. Как объявить одномерный массив?
31. Что такое индекс массива? С какого числа он может начинаться?
32. Как обратиться к (i) -тому элементу массива?
33. Как осуществляется ввод и вывод элементов массива?
34. Назовите основные алгоритмы обработки массивов (сумма, произведение, поиск максимума/минимума).
35. Что такое двумерный массив (матрица)? Как к нему обратиться?
36. Что такое строковый тип данных (string)? Как работать с символами в строке?
37. Что такое подпрограмма? Какие два типа подпрограмм вы знаете (процедура/функция)?
38. Чем функция отличается от процедуры?
39. Что такое параметры подпрограммы? Чем отличаются формальные параметры от фактических?
40. Как передать значение в функцию? Что такое возвращаемое значение?
41. Что такое рекурсия? Приведите пример рекурсивной функции.
42. Какова роль библиотек и модулей в программировании?
43. Что такое структурное программирование?.
44. В чем заключается задача поиска элемента в массиве?
45. Как работает алгоритм линейного поиска?
46. Как работает алгоритм бинарного поиска? Какое условие должно выполняться для массива?.
47. В чем заключается задача сортировки массива?



48. Опишите принцип работы сортировки пузырьком (Bubble sort).
49. Чем отличается сортировка выбором от сортировки пузырьком?
50. Что такое «сложность алгоритма»

Типовые практические задания для оценки знаний и умений промежуточной аттестации

1. Составить программу линейной структуры.
2. Составить программу разветвляющейся структуры.
3. Составить программу циклической структуры
4. Составить программу поиска элемента массива.
5. Составить программу сортировки массива.
6. Разработать функцию.
7. Разработать класс.
8. Составить программу с использованием указателей или ссылок.
9. Составить программу, используя для форматирования вывода компонентные функции класса ios.

На экзамене студент отвечает на теоретический вопрос и выполняет практическое задание по изученному материалу.

Студент создает компьютерные файлы, сохраняя их на диске под своей фамилией. Выполненное задание оценивается по пятибалльной системе.

4.4. Критерии и показатели оценивания

Для текущего контроля

| Оценка | Форма контроля | Критерии оценивания | Показатели оценивания |
|---------------|----------------------------------|---|---|
| «5» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено правильно в полном объеме в соответствии с выданным заданием |
| «4» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено правильно в полном объеме в соответствии с выданным заданием, имеются 1-2 ошибки исправленные самостоятельно по просьбе преподавателя |
| «3» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено более чем на 50% |
| «2» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено менее чем на 50% или не выполнено вовсе. |

Критерии оценки теста

Результаты оцениваются следующим образом: «неудовлетворительно» <50 % правильных ответов; «удовлетворительно» 50-69 % правильных ответов; «хорошо» 69-85 % правильных ответов; «отлично» > 85 % правильных ответов.

Для промежуточной аттестации



| Оценка | Форма контроля | Критерии оценивания | Показатели оценивания |
|--------|----------------------------------|---|---|
| «5» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено правильно в полном объеме в соответствии с выданным заданием |
| «4» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено правильно в полном объеме в соответствии с выданным заданием, имеются 1-2 ошибки исправленные самостоятельно по просьбе преподавателя |
| «3» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено более чем на 50% |
| «2» | Выполнение практического задания | полнота и правильность выполнения задания | Задание выполнено менее чем на 50% или не выполнено вовсе. |

| Оценка | Форма контроля | Критерии оценивания | Показатели оценивания |
|--------|----------------|---|--|
| «5» | устный ответ | полнота и правильность ответа, степень осознанности, понимания изученного материала, четкость и грамотность речи. | ответ полный и правильный на основании изученных теорий; материал изложен в определенной логической последовательности, литературным языком; ответ самостоятельный. |
| «4» | устный ответ | полнота и правильность ответа, степень осознанности, понимания изученного материала, четкость и грамотность речи. | ответ полный и правильный на основании изученных теорий; материал изложен в определенной логической последовательности, при этом допущены две-три несущественные ошибки, исправленные по требованию учителя. |
| «3» | устный ответ | полнота и правильность ответа, степень осознанности, понимания изученного материала, четкость и грамотность речи. | ответ полный, но при этом допущена существенная ошибка, или неполный, несвязный. |
| «2» | устный ответ | полнота и правильность ответа, степень осознанности, понимания изученного материала, четкость и грамотность речи. | при ответе обнаружено непонимание учащимся основного содержания учебного материала или допущены существенные ошибки, которые учащийся не смог исправить при наводящих |



| | | | |
|--|--|--|-------------------|
| | | | вопросах учителя. |
|--|--|--|-------------------|

5. Условия реализации программы дисциплины

5.1. Для реализации программы учебной дисциплины должны быть предусмотрены следующие специальные помещения:

Кабинет общепрофессиональных дисциплин и профессиональных модулей

| | | | | |
|---|--|--------|----------|---|
| 2 | Рабочее место преподавателя | Мебель | Основное | На усмотрение ОО |
| 3 | Шкаф или полки для хранения учебной и методической литературы | Мебель | Основное | На усмотрение ОО |
| 4 | Доска маркерная | Мебель | Основное | На усмотрение ОО |
| 5 | ПК преподавателя (системный блок, монитор, клавиатура, мышь) | ТС | Основное | ЦПУ: - Intel(R) Core(TM) i3-10100 - количество физических ядер - 4 - количество потоков - 8 Сетевой адаптер: - технология Ethernet - 10/100/1000 mbps ОЗУ: - 8 ГБ Графический адаптер: - NVIDIA GeForce GT730 ПЗУ:- SSD 256 ГБ |
| 6 | ПК (системный блок, монитор, клавиатура, мышь) по количеству обучающихся | ТС | Основное | ЦПУ: - Intel(R) Core(TM) i3-10100 - количество физических ядер - 4 - количество потоков - 8 Сетевой адаптер: - технология Ethernet - 10/100/1000 mbps ОЗУ: - 8 ГБ Графический адаптер: - NVIDIA GeForce GT730 ПЗУ:- SSD 256 ГБ |
| 7 | Мультимедийный проектор | ТС | Основное | На усмотрение ОО |
| 8 | Аудио- и видеооборудование | ТС | Основное | На усмотрение ОО |
| 9 | Комплект учебно-методических материалов | УМК | Основное | На усмотрение ОО |

| № п/п | Наименование лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства |
|-------|---|
| 1 | Операционные системы для обеспечения функционирования программных средств общего |



| | |
|----|--|
| | и профессионального обозначения на рабочих местах преподавателей и обучающихся |
| 2 | Пакет стандартных офисных приложений для работы с документами, таблицами, базами данных и т.п. |
| 3 | Программы просмотра текстовых и графических документов |
| 4 | Программы-архиваторы |
| 5 | Интернет-браузеры (не менее двух) |
| 6 | Антивирусные программы (не менее двух) |
| 7 | Программы для восстановления данных и файлов |
| 8 | Интегрированные среды разработки программного обеспечения: Microsoft Visual Studio, Android Studio, Java SE Development Kit, Arduino IDE или аналогичные |
| 9 | Microsoft Visio или аналогичная |
| 10 | OTRS/ osTicket, Boas Help Desk/ Liberum Help Desk и/или подобные системы |
| 11 | Okdesk, HelpDeskEddy, ITSM 365, IntraService, Service Creatio, HubEx, Omnidesk, Happydesk, Кауако и/или подобные системы. |
| 12 | Средства автоматизированного проектирования Компас, Autocad Eagle (Fusion360), NI Multisim, Cadence Allegro Platform или аналогичные |

6. Информационное обеспечение реализации программы

Для реализации программы библиотечный фонд образовательной организации, преподаватели, студенты имеют доступ к электронной библиотечной системе Znanium.com.

6.1. Основные печатные издания

2. Гуриков, С.Р. Основы алгоритмизации и программирования на Visual C++ : учебное пособие / С.Р. Гуриков. — Москва : ИНФРА-М, 2022. — 515 с. — (Высшее образование: Бакалавриат). — DOI 10.12737/1039154.

Режим доступа: <https://znanium.com/read?id=397332>

3. Объектно-ориентированное программирование на C++ : учебник / И. В. Баранова, С. Н. Баранов, И. В. Баженова [и др.]. — Красноярск : Сиб. федер. ун-т, 2019. — 288 с. ISBN 978-5-7638-4034-6 Режим доступа: <https://znanium.com/read?id=380554>

6.3. Дополнительные источники

4. Игнашева, Е.П. Системы счисления, алгоритмизация и программирование : учебное пособие / Е.П. Игнашева. — Москва : ИНФРА-М, 2023. — 224 с. — (Военное образование).

Режим доступа: <https://znanium.com/read?id=426189>

5. Гуриков, С.Р. Основы алгоритмизации и программирования на языке Microsoft Visual Basic : учебное пособие / С.Р. Гуриков. — Москва : ИНФРА-М, 2022. — 594 с. — (Среднее профессиональное образование).

Режим доступа: <https://znanium.com/read?id=397645>

4. Научно-технический и научно-производственный журнал «Информационные технологии» <http://novtex.ru/IT/index.htm>

6. Журнал «Бизнес-информатика» <https://bijournal.hse.ru/>

7. Журнал «Информационные системы и технологии» <http://oreluniver.ru/science/journal/isit>

8. Журнал «Электронные информационные системы»