

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
«Национальный исследовательский ядерный университет «МИФИ»  
**Новоуральский технологический институт**  
филиал федерального государственного автономного образовательного учреждения высшего  
образования «Национальный исследовательский ядерный университет «МИФИ»  
**(НТИ НИЯУ МИФИ)**  
**Колледж НТИ**

---

Цикловая методическая комиссия информационных технологий

ОДОБРЕНО

Учёным Советом НТИ НИЯУ МИФИ

Протокол № 5 от 02 сентября 2025г.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
**ПО УЧЕБНОЙ ДИСЦИПЛИНЕ**  
**ОП.03 «ОСНОВЫ АЛГОРИТМИЗАЦИИ И**  
**ПРОГРАММИРОВАНИЯ»**

для студентов колледжа НТИ НИЯУ МИФИ,  
обучающихся по программе среднего профессионального образования

специальность 09.02.09

«Веб-разработка»

очная форма обучения

на базе основного общего образования

квалификация


разработчик веб-приложений

Новоуральск 2025

ОДОБРЕНО:  
на заседании  
цикловой методической комиссии  
информационных технологий  
Протокол № 8 от 01.09.2025 г.

Составлен в соответствии с рабочей программой учебной дисциплины ОП.03 «Основы алгоритмизации и программирования» по специальности 09.02.09 «Веб-разработка»

Председатель ЦМК ИТ

 И.И. Горницкая

Фонд оценочных средств по учебной дисциплине ОП.03 «Основы алгоритмизации и программирования» – Новоуральск: Изд-во колледжа НТИ НИЯУ МИФИ, 2025. – 26с.

## АННОТАЦИЯ

Фонд оценочных средств предназначен для текущего контроля и промежуточной аттестации обучающихся по специальности 09.02.09 «Веб-разработка» на соответствие их персональных достижений поэтапным требованиям программы подготовки специалистов среднего звена по учебной дисциплине ОП.03 «Основы алгоритмизации и программирования». Комплектация фонда оценочных средств: паспорт, программа оценивания, оценочные средства для текущего контроля и промежуточной аттестации по учебной дисциплине, критерии оценивания. В паспорте фонда оценочных средств указаны: место учебной дисциплины в структуре программы подготовки специалистов среднего звена, требования ФГОС СПО к результатам освоения учебной дисциплины, перечень формируемых компетенций, компоненты фонда оценочных средств

Разработчик: Лебедева А.Н., преподаватель ЦМК информационных технологий

Редактор: Горницкая И.И.

## СОДЕРЖАНИЕ

ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ ОП.03 «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ» .....	4
ПРОГРАММА ОЦЕНИВАНИЯ.....	7
ОЦЕНОЧНЫЕ СРЕДСТВА И КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ.....	10
ОЦЕНОЧНЫЕ СРЕДСТВА И КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ.....	22

**ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ  
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ  
ОП.03 «ОСНОВЫ АЛГОРИТМИЗАЦИИ И  
ПРОГРАММИРОВАНИЯ»**

Фонд оценочных средств является составной частью учебно-методических документов, обеспечивающих реализацию основной образовательной программы СПО по специальности 09.02.09 Веб-разработка.

Фонд оценочных средств предназначен для проверки результатов освоения учебной дисциплины ОП.03 «Основы алгоритмизации и программирования».

Место дисциплины в структуре основной образовательной программы: учебная дисциплина ОП.03 «Основы алгоритмизации и программирования» принадлежит к общепрофессиональному циклу.

Цель и планируемые результаты освоения дисциплины:

<b>Код ПК, ОК</b>	<b>Умения</b>	<b>Знания</b>
ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	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	Понятия алгоритмизации, свойства алгоритмов, общих принципов построения алгоритмов, основных алгоритмических конструкции; Эволюции языков программирования, их классификации, понятие системы программирования; Основных элементов языка, структуры программы, операторов и операций, управляющих структур, структур данных, файлов, классов памяти;

	<p>программирования;  Оформлять код программы в соответствии со стандартом кодирования.  Выполнять проверку, отладку кода программы</p>	<p>Понятие подпрограммы, составление библиотек подпрограмм;  Объектно-ориентированной модели программирования, основных принципов объектно-ориентированного программирования: понятие классов и объектов, их свойств и методов, инкапсуляция и полиморфизма, наследования и переопределения</p>
--	---	---

Перечень формируемых компетенций в соответствии с требованиями ФГОС СПО:

Общие компетенции (ОК):

ОК 01 Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам.

ОК 02 Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности.

ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста.

ОК 07 Содействовать сохранению окружающей среды, ресурсосбережению, применять знания об изменении климата, принципы бережливого производства, эффективно действовать в чрезвычайных ситуациях

Профессиональные компетенции (ПК):

ПК 1.1 Проектировать информационные ресурсы

ПК 1.3 Интегрировать программный код в соответствующую инфраструктуру

ПК 1.5 Выполнять процедуры тестирования программного кода

ПК 2.2 Проводить работы по резервному копированию и развертыванию резервной копии информационных ресурсов

ПК 2.3 Настраивать права пользователей в соответствии с функциональными задачами (ролями) и на основании информации о поведенческих факторах

ПК 2.4 Применять программные средства обеспечения безопасности информации веб-приложений

Фонд оценочных средств по учебной дисциплине ОП.03 «Основы алгоритмизации и программирования» включает оценочные средства для текущего контроля и оценочные средства для проведения промежуточной аттестации.

## ПРОГРАММА ОЦЕНИВАНИЯ

№ п/п	Контролируемые разделы, темы учебной дисциплины	Контролируемые компетенции (или их части)	Вид оценивания
1	2	3	4
1	Тема 1.1 Алгоритмизация	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)
2	Тема 1.2 Основы технологии программирования	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)
3	Тема 2.1 Алфавит языка программирования. Типы данных	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)
4	Тема 2.2 Операторы языка	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)
5	Тема 2.3 Массивы	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2,	Устный опрос

<b>№ п/п</b>	<b>Контролируемые разделы, темы учебной дисциплины</b>	<b>Контролируемые компетенции (или их части)</b>	<b>Вид оценивания</b>
		ПК 2.3, ПК 2.4	
6	Тема 2.4 Строки	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос
7	Тема 2.5 Пользовательские типы данных	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)
8	Тема 2.6 Функции	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)
9	Тема 2.7 Работа с файлами	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)
10	Тема 2.8 Динамические структуры данных	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Устный опрос Письменный опрос (тестирование)

№ п/п	Контролируемые разделы, темы учебной дисциплины	Контролируемые компетенции (или их части)	Вид оценивания
11	Практические занятия	ОК 01, ОК 02, ОК 05, ОК 07, ПК 1.1, ПК 1.3, ПК 1.5, ПК 2.2, ПК 2.3, ПК 2.4	Представление решения профессиональных задач, защита отчета по практическому занятию
12	Промежуточная аттестация по учебной дисциплине	IV семестр промежуточная аттестация в форме экзамена (с консультацией и подготовкой к экзамену)	

# **ОЦЕНОЧНЫЕ СРЕДСТВА И КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ**

Текущий контроль проводится на теоретических и практических занятиях и включает в себя оценку знаний и умений, компетенций обучающихся.

Формы проведения текущего контроля: устный опрос, письменный опрос (может быть проведен в форме тестирования), выполнение практических работ на практических занятиях.

## **Примерный перечень вопросов для проведения устного опроса**

### **Тема 1.1 Алгоритмизация**

- 1 Что такое алгоритм? Приведите примеры алгоритмов из повседневной жизни.
- 2 Перечислите и охарактеризуйте основные свойства алгоритмов.
- 3 Какие существуют способы описания алгоритмов?
- 4 Что такое блок-схема? Опишите основные элементы блок-схем.
- 5 Расскажите о линейной алгоритмической конструкции.
- 6 Объясните принцип работы разветвляющейся алгоритмической конструкции.
- 7 Опишите циклическую алгоритмическую конструкцию.
- 8 В чем заключается метод пошаговой детализации алгоритмов?
- 9 Как определяется сложность алгоритма?
- 10 Приведите примеры задач, решаемых с помощью различных алгоритмических конструкций.
- 11 Что такое трассировка алгоритма?
- 12 Какие существуют способы доказательства правильности алгоритма?
- 13 Опишите основные этапы разработки алгоритма.
- 14 В чем отличие детерминированного алгоритма от недетерминированного?

15 Что такое вычислительная сложность алгоритма?

### **Тема 1.2 Основы технологии программирования**

1. Что такое «технология программирования» и чем она отличается от просто написания кода?
2. Назовите основные составляющие технологии программирования.
3. Что является основной идеей структурного программирования?
4. Какие три основные управляющие конструкции лежат в основе структурного программирования?
5. Почему в структурном программировании не рекомендуется использовать оператор безусловного перехода goto?
6. Что такое «модуль» в контексте модульного программирования?
7. Каковы главные цели разбиения программы на модули?
8. Что означает принцип «сокрытия информации» в модульном программировании?
9. В чем основное концептуальное отличие объектно-ориентированного программирования от структурного?
10. Назовите три основных принципа объектно-ориентированного программирования.
11. Что такое «инкапсуляция» и какую проблему она помогает решить?
12. Объясните, что такое «наследование» и как оно способствует повторному использованию кода.
13. Что такое «полиморфизм»? Приведите простой пример.
14. Как развитие технологий программирования повлияло на сложность создаваемых программ?
15. Какую парадигму программирования вы бы выбрали для разработки простой утилиты для обработки текста и почему? А для большой банковской системы?

### **Тема 2.1 Алфавит языка программирования. Типы данных**

- 1 Что такое идентификатор в языке программирования?
- 2 Расскажите о правилах именовании переменных.

- 3 Перечислите основные типы данных в языках программирования.
- 4 В чем разница между целочисленными и вещественными типами данных?
- 5 Что такое литерал? Приведите примеры.
- 6 Опишите операции присваивания.
- 7 Расскажите о стандартных операциях ввода-вывода.
- 8 Что такое константа? Приведите примеры.
- 9 Как объявляются переменные различных типов?
- 10 В чем отличие явного и неявного преобразования типов?
- 11 Какие существуют символьные типы данных?
- 12 Опишите логический тип данных.
- 13 Что такое массив как тип данных?
- 14 Как осуществляется инициализация переменных?
- 15 Расскажите о правилах приоритета операций.

### **Тема 2.2 Операторы языка**

- 1 Что такое условный оператор? Приведите синтаксис.
- 2 Опишите работу оператора множественного выбора.
- 3 Расскажите о различных видах циклов.
- 4 В чем отличие цикла с предусловием от цикла с постусловием?
- 5 Как работают вложенные циклы?
- 6 Что такое оператор перехода?
- 7 Опишите составной оператор.
- 8 Как осуществляется досрочный выход из цикла?
- 9 Расскажите о работе оператора возврата.
- 10 Что такое бесконечный цикл?
- 11 Как реализуются итерационные циклы?
- 12 Опишите обработку прерываний в циклах.
- 13 В чем особенность вложенных условных операторов?
- 14 Как работают операторы сравнения?
- 15 Расскажите о логических операторах.

## **Тема 2.3 Массивы**

- 1 Что такое одномерный массив?
- 2 Как осуществляется ввод элементов массива?
- 3 Опишите методы обработки одномерных массивов.
- 4 Что такое двумерный массив?
- 5 Как организуется доступ к элементам двумерного массива?
- 6 Расскажите о методах сортировки массивов.
- 7 Как осуществляется поиск элементов в массиве?
- 8 Опишите алгоритмы работы с подмассивами.
- 9 Что такое динамический массив?
- 10 Как реализуется изменение размера массива?
- 11 Расскажите о многомерных массивах.
- 12 Как организуется передача массивов в функции?
- 13 Опишите алгоритмы работы с матрицами.
- 14 Что такое разреженный массив?
- 15 Как осуществляется сериализация массивов?

## **Тема 2.4 Строки**

- 1 Что такое строка в программировании?
- 2 Перечислите основные операции, которые можно выполнять со строками.
- 3 Как осуществляется конкатенация строк в выбранном языке программирования?
- 4 Объясните, что такое индексация строк.
- 5 Что такое срезы строк?
- 6 Расскажите о методах поиска подстроки в строке.
- 7 Опишите способы преобразования регистра символов в строке.
- 8 Как удалить или заменить часть строки?
- 9 Что такое форматирование строк?
- 10 Как проверить, состоит ли строка только из цифр, букв, пробелов и т.п.? Опишите соответствующие строковые методы и приведите примеры.

11 Объясните, как работают регулярные выражения для работы со строками.

12 Опишите способы разбиения строки на подстроки.

13 Что такое экранирование символов в строках?

14 В чём разница между обычными строками и сырыми строками.

15 Опишите особенности работы со строками в Unicode.

### **Тема 2.5 Пользовательские типы данных**

1. Что такое структура данных?

2. Как объявляются пользовательские типы?

3. Опишите работу с записями.

4. Что такое перечисление?

5. Как создаются объединения?

6. Расскажите о вложенных типах данных.

7. Как осуществляется доступ к полям структуры?

8. Опишите методы работы с записями.

9. Опишите, как создать структуру. Приведите пример объявления структуры для хранения данных о студенте (ФИО, номер зачётной книжки, средний балл).

10. В чём разница между структурой и классом? Сравните их с точки зрения доступа к членам и использования в программах.

11. Как создать собственный тип данных на основе существующего?

12. Опишите концепцию объединения. В чём его основное отличие от структуры?

13. Что такое класс как пользовательский тип данных? Опишите основные компоненты класса

14. Приведите краткий пример класса «Автомобиль».

15. Как организовать массив пользовательских типов данных?

### **Тема 2.6 Функции**

1. Что такое функция в программировании?

2. Какие существуют типы функций?

3. Как происходит передача параметров в функцию?
4. В чем разница между формальными и фактическими параметрами?
5. Как осуществляется возврат значения из функции?
6. Что такое рекурсивная функция?
7. Какие бывают виды рекурсии?
8. Как происходит обработка ошибок в функциях?
9. Что такое перегрузка функций?
10. Как работает механизм области видимости в функциях?
11. Какие существуют способы оптимизации функций?
12. Как происходит инлайнинг функций?
13. Что такое чистые функции?
14. Как работают функции с переменным числом параметров?
15. Какие существуют соглашения о вызовах функций?

### **Тема 2.7. Работа с файлами**

1. Что такое файл в программировании?
2. Какие существуют типы файлов?
3. Как происходит открытие файла?
4. Какие режимы доступа к файлу существуют?
5. Как осуществляется чтение данных из файла?
6. Как происходит запись данных в файл?
7. Что такое позиционирование в файле?
8. Как осуществляется поиск в файле?
9. Какие существуют методы обработки ошибок при работе с файлами?
10. Как происходит создание и удаление файлов?
11. Что такое буферизация при работе с файлами?
12. Как осуществляется работа с бинарными файлами?
13. Какие существуют способы сортировки файлов?
14. Как происходит индексация файлов?
15. Что такое потоковый ввод-вывод?

## **Тема 2.8. Динамические структуры данных**

1. Что такое динамическая структура данных?
2. Как работает стек?
3. Какие операции выполняются над стеком?
4. Как реализуется очередь?
5. Какие существуют виды очередей?
6. Что такое связный список?
7. Как работает односвязный список?
8. Как реализуется двусвязный список?
9. Что такое циклический список?
10. Как происходит вставка и удаление элементов в динамических структурах?
11. Какие существуют способы реализации деревьев?
12. Как работает граф в памяти?
13. Что такое хеш-таблица?
14. Как происходит управление памятью при работе с динамическими структурами?
15. Какие существуют алгоритмы обхода динамических структур?

### **Критерии оценивания ответов на вопросы для проведения устного опроса**

«ОТЛИЧНО». Обучающийся демонстрирует глубокое и прочное усвоение материала темы. Дан полный, развернутый ответ на вопрос, показана совокупность осознанных знаний по дисциплине, доказательно раскрыты основные положения; в ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, теорий, явлений. Знание по дисциплине демонстрируется на фоне понимания его в системе данной науки и междисциплинарных связей. Ответ изложен литературным языком с использованием методической терминологии. Могут быть допущены недочеты в определении понятий, исправленные студентом самостоятельно в процессе ответа.

«ХОРОШО». Обучающийся демонстрирует знания в объеме пройденной программы. Дан полный, развернутый ответ на поставленный вопрос, показано умение выделить существенные и несущественные признаки, причинно-следственные связи. Ответ четко структурирован, логичен, изложен литературным языком. Могут быть допущены 2-3 неточности или незначительные ошибки, исправленные студентом с помощью дополнительных вопросов и уточняющих вопросов преподавателя.

«УДОВЛЕТВОРИТЕЛЬНО». Обучающийся демонстрирует недостаточно полное усвоение знаний по пройденной программе. Дан недостаточно полный и недостаточно развернутый ответ. Логика и последовательность изложения имеют нарушения. Допущены ошибки в раскрытии понятий, употреблении терминов. Студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи. Умение раскрыть значение обобщенных знаний не показано. Речевое оформление требует поправок, коррекции.

«НЕУДОВЛЕТВОРИТЕЛЬНО». Обучающийся демонстрирует непонимание основного содержания учебного материала. Ответ представляет собой разрозненные знания с существенными ошибками по вопросу. Присутствуют фрагментарность, нелогичность изложения. Отсутствуют конкретизация и доказательность изложения. Речь неграмотная. Дополнительные и уточняющие вопросы преподавателя не приводят к коррекции ответа студента.

## **Перечень заданий для проведения письменного опроса**

### **Тестирование**

Форма проведения письменного опроса – тестирование.

Тестирование проводится на образовательной платформе для университетов и колледжей, содержащей цифровой учебный контент и сервисы для эффективного образования – <https://urait.ru/>

Учебник: Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для среднего профессионального образования / Д.

Ю. Федоров. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 227 с. — (Профессиональное образование). — ISBN 978-5-534-17319-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. с. 174 — URL:<https://urait.ru/bcode/539652/p.174>

Чек-лист:

- 1 Войти на платформу Юрайт
- 2 Пройти тестирование по ссылке
- 3 Отправить результаты прохождения теста преподавателю на платформу Юрайт

### **Тема 1.1 Алгоритмизация**

<https://urait.ru/quiz/run-test/BADAC700-F311-40AD-A385-FA2075522733/B044CE82-28E6-450A-932D-3166BD589348>

### **Тема 1.2 Основы технологии программирования**

<https://urait.ru/quiz/run-test/9CE29268-ED4A-4B52-B6DB-F4549FACD2B1/CA6024D4-ECE0-4B1F-929C-AC6DFC1E4DA3>

### **Тема 2.1 Алфавит языка программирования. Типы данных**

<https://urait.ru/quiz/run-test/8CFE1E30-9C59-4911-B732-CB9C700DB112/4924975E-535E-44B9-ADDE-55E1BE013D47>

### **Тема 2.2 Операторы языка**

<https://urait.ru/quiz/run-test/C9C4DB51-B9C2-463D-988C-1FDB576B95E5/8792A9E2-4194-49DD-87DF-87F8354710BE>

<https://urait.ru/quiz/run-test/F544155E-B537-4BE1-B27D-D550F7A9AB03/467E9A98-9BEE-4703-A549-7968106F9C9F>

### **Тема 2.5 Пользовательские типы данных**

<https://urait.ru/quiz/run-test/A1C48662-B22F-4B93-B641-2B7ABD63954F/ED760669-5483-4829-B94B-F0C55FE9BAC5>

### **Тема 2.6 Функции**

<https://urait.ru/quiz/run-test/0168C8B8-1630-4439-A86D-3F700C57C839/25E8E2C1-6A66-43F8-B197-DF888584364E>

### **Тема 2.7 Работа с файлами**

<https://urait.ru/quiz/run-test/DD5623BC-01C9-404D-91D1-801ECD3B852C/69E9283C-5499-49A7-A41D-B268CBD37DC5>

## **Тема 2.8 Динамические структуры данных**

<https://urait.ru/quiz/run-final-test/C7CE975C-8593-4F47-965C-6B34D1C57A67>

### **Критерии оценки результатов выполнения тестовых заданий**

Для интерпретации результатов тестирования используется переводная шкала, с помощью которой результаты теста преобразуют в оценку знаний студентов по пятибалльной системе.

«ОТЛИЧНО» – 90%-100% верных ответов

«ХОРОШО» – 75%-89% верных ответов,

«УДОВЛЕТВОРИТЕЛЬНО» – 74%-60% верных ответов

«НЕУДОВЛЕТВОРИТЕЛЬНО» – 59%-0% верных ответов

### **Практические занятия**

Учебным планом предусмотрено проведение практических занятий:

Практическое занятие №1. Разработка алгоритмов линейной, разветвляющейся и циклической структуры

Практическое занятие № 2. Разработка программ линейной структуры

Практическое занятие № 3. Разработка программ разветвляющейся структуры  
Практическое занятие № 4. Разработка программ циклической структуры

Практическое занятие № 5. Разработка программ с использованием одномерных массивов

Практическое занятие № 6. Разработка программ с использованием двумерных массивов

Практическое занятие № 7. Разработка программ с использованием стандартных функций для работы со строками и массивами

Практическое занятие № 8. Разработка программ с использованием пользовательских типов данных

Практическое занятие № 9. Разработка функций с использованием одномерных массивов

Практическое занятие № 10. Разработка функций с использованием двумерных массивов

Практическое занятие № 11. Разработка программ с использованием рекурсии  
Практическое занятие № 12. Разработка функций с использованием данных строкового типа

Практическое занятие № 13. Создание файла. Чтение из файла. Изменение данных в файле

Практическое занятие № 14. Разработка программ с использованием однонаправленных списков типа «стек»

Практическое занятие № 15. Разработка программ с использованием однонаправленных списков типа «очередь».

Практическое занятие № 16. Разработка программ с использованием двусвязных списков

Практическое занятие № 17. Разработка алгоритмов с использованием динамических структур данных: очередей и стеков.

### **Критерии оценивания результатов выполнения практических работ и подготовки отчета по практическому занятию**

Критериями оценки результативности практического занятия являются:

- степень реализации цели и задач работы;
- степень выполнения заданий;
- степень соответствия результатов работы заданным требованиям;
- качество подготовки отчета;
- степень сформированности у студентов необходимых умений и навыков.

«ОТЛИЧНО». Обучающийся выполняет профессиональные действия и демонстрирует практические умения без ошибок, в полной мере владеет учебным материалом, самостоятельно интерпретирует полученные результаты, технически грамотно формулирует выводы. Не допускает

ошибок в процессе защиты отчёта. Отчёт оформлен в соответствии с установленными требованиями.

«ХОРОШО». Обучающийся выполняет профессиональные действия и демонстрирует практические умения с небольшими недочетами, в полной мере владеет учебным материалом, контрольные вопросы содержат недостатки, технически грамотно формулирует выводы. Задания для самостоятельного выполнения осуществляет с незначительными ошибками. Допускает незначительные ошибки в процессе защиты отчёта. Отчёт оформлен в соответствии с установленными требованиями;

«УДОВЛЕТВОРИТЕЛЬНО». Обучающийся выполняет профессиональные действия и демонстрирует практические умения с недочетами, слабо владеет учебным материалом, контрольные вопросы содержат ошибки, неграмотно формулирует выводы. Задания для самостоятельного выполнения осуществляет с ошибками. Допускает ошибки в процессе защиты отчёта. Отчёт оформлен в соответствии с установленными требованиями;

«НЕУДОВЛЕТВОРИТЕЛЬНО». Обучающийся выполняет профессиональные действия и демонстрирует практические умения со значительными ошибками, не владеет учебным материалом, контрольные вопросы содержат ошибки, неграмотно формулирует выводы. Задания для самостоятельного выполнения осуществляет неправильно. Допускает ошибки в процессе защиты отчёта. Отчёт оформлен не в соответствии с установленными требованиями.

# **ОЦЕНОЧНЫЕ СРЕДСТВА И КРИТЕРИИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ДЛЯ ПРОВЕДЕНИЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**

## **Экзамен**

Промежуточная аттестация по учебной дисциплине ОП.03 «Основы алгоритмизации и программирования» проводится в форме экзамена.

До даты проведения экзамена необходимо пройти оценивание всех работ, выполненных в течение IV семестра (2 курс обучения): теоретическое обучение, практические занятия.

Оценки экзамена – отлично, хорошо, удовлетворительно, неудовлетворительно.

### **Пример типового экзаменационного билета**

- 1 Задание 1: теоретический блок
- 2 Задание 2: постановка задачи для разработки программы, разработка блок-схемы программы, написание текста программы с комментариями
- 3 Задание 3: для разработанной программы создать сценарий и набор для тестирования, провести анализ эффективности разработанного кода.

### **Примерный перечень вопросов для проведения экзамена**

- 1 Что такое алгоритм?
- 2 Какие существуют свойства алгоритма?
- 3 Назовите основные типы алгоритмов.
- 4 Чем отличается цикл с предусловием от цикла с постусловием?
- 5 Расскажите о способах описания алгоритмов.
- 6 Какой принцип используется в структурированном программировании?
- 7 Охарактеризуйте понятие «модуль» в программировании.
- 8 Какие этапы включает процесс проектирования программного продукта?
- 9 Какие бывают уровни языков программирования?
- 10 Чем отличаются компиляторы от интерпретаторов?

- 11 Что означает термин «локальная переменная»?»?
- 12 Назовите основные операторы управления потоком исполнения в языках программирования.
- 13 Определите понятие «класс памяти».
- 14 Что такое глобальные переменные и почему их использование ограничено?
- 15 Почему важно соблюдать стандарты оформления кода?
- 16 Что такое объектно-ориентированное программирование?
- 17 В чём разница между процедурой и функцией?
- 18 Назовите базовые понятия объектно-ориентированного программирования.
- 19 Что такое полиморфизм?
- 20 Расскажите о понятии абстрактного класса.
- 21 В чём заключается механизм наследования?
- 22 Назовите особенности рекурсивных функций.
- 23 Что такое графическое представление алгоритма?
- 24 Какие структуры данных используются для хранения упорядоченной последовательности данных?
- 25 Что такое стек и очередь? В чём различие между ними?
- 26 Зачем нужны библиотеки стандартных функций?
- 27 Как устроены файлы в программах на уровне программирования?
- 28 Какие подходы применяются для анализа сложности алгоритмов?
- 29 В чём отличие Java от других популярных языков программирования?
- 30 Чем обусловлена популярность Python в научном сообществе и сфере машинного обучения?

### **Примерный перечень практических заданий для проведения экзамена**

#### **Варианты постановок задач для разработки программ**

- 1 Разработать программу для игры. «Описание игры. На экране появляется вопрос: «Чет (введите 2) или нечет (введите 1)?» Играющий

должен ответить, какое число – четное или нечетное – выберет компьютер, и ввести соответственно 2 или 1. После этого компьютер случайным образом генерирует одно из них. Результат сравнения ответа играющего с числом компьютера выводится на экран.»

2 Разработать игру, моделирующую бросание игрового кубика каждым из двух участников, после чего определяется у кого, выпало больше очков.

3 Разработать игру «Описание игры. Компьютер генерирует случайное целое число, большее 0 и меньше 101. Играющий пытается отгадать это число, делая несколько попыток. В случае несовпадения «задуманного» компьютером числа и числа-ответа на экран выводится сообщение о том, какое из них больше, после чего играющий вновь вводит число, и т. д. до отгадывания»

4 Разработать программу-калькулятор, способную выполнять базовые арифметические операции (сложение, вычитание, умножение, деление). Добавьте проверку ввода данных и обработку ошибок деления на ноль.

5 Разработать программу организации списка задач (To-do List). Простая система для управления списком дел студента. Студент сможет добавлять задачи, устанавливать сроки выполнения, отмечать выполненные задачи и фильтровать задачи по статусу («выполнено», «не выполнено»).

6 Разработать игру «Игра моделирует выбор каждым из двух играющих «наугад» по одной карте из полного набора игральных карт, включающего четыре масти («пики», «трефы», «бубны» и «червы») и по 9 достоинств карт в каждой масти («шестерка», «семерка», «восьмерка», «девятка», «десятка», «валет», «дама», «король», «туз»), и определение того из участников игры, у которого выбранная карта «старше». При этом условимся, что приведенный выше перечень мастей и карт одной масти дан в порядке увеличения их «старшинства» (например, любая карта масти «бубны» старше любой карты масти «пики», а «валет червей» старше «десятки червей»))»

7 Разработать программу телефонной книги, позволяющую добавлять, удалять и искать контакты по имени или номеру телефона. Реализовать сортировку контактов по алфавиту.

8 Разработать программу, которая принимает предложение и выводит частоту появления каждой буквы алфавита в нём.

9 Разработать консольное приложение игры «Камень-Ножницы-Бумага», которое позволяет играть против компьютера. Обеспечить возможность выбора уровня сложности и отображение статистики побед и поражений.

10 Разработать программу-конвертер физических единиц измерения (например, длины, массы, температуры). Предоставить возможность пользователям выбирать единицы измерения и вводить значения для преобразования.

11 Разработать игру «Змейка» с классическими правилами. Добавить поддержку клавиатуры для управления змейкой, визуализацию игрового поля и подсчет очков игрока.

12 Разработать игру «Описание игры. На столе выложены  $n$  предметов (спичек, монет, камешков и т. п.). Играют двое. Они поочередно забирают несколько предметов, причем заранее договорено, что число забранных предметов не превышает  $k$  ( $1 < k \leq n$ ). Проигрывает тот, кто своим ходом вынужден забрать последний предмет».

### **Критерии оценивания знаний обучающихся на экзамене**

«ОТЛИЧНО». Представлен развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. Обучающийся ориентируется в излагаемом материале, отвечает на дополнительные вопросы, демонстрирует глубокие теоретические знания, знание первоисточников.

«ХОРОШО». Представлен достаточно развернутый ответ на теоретический вопрос, а также поэтапное решение практического задания с пояснениями. В решении практического задания могут быть допущены

ошибки, не искажающие лежащего в основе технического решения алгоритма. Обучающийся уверенно отвечает на дополнительные вопросы, демонстрирует достаточно высокий уровень теоретических знаний, знание первоисточников.

«УДОВЛЕТВОРИТЕЛЬНО». Представлен неполный ответ на теоретический вопрос. В решении практического задания могут быть допущены ошибки, не искажающие лежащего в основе технического решения алгоритма. Обучающийся демонстрирует достаточный уровень теоретических знаний, однако затрудняется отвечать на отдельные вопросы.

«НЕУДОВЛЕТВОРИТЕЛЬНО». Теоретический вопрос не раскрыт, в решении практического задания допущены существенные ошибки, ввиду незнания алгоритмов технического решения. Либо дан ответ только на один из вопросов билета. Обучающийся затрудняется отвечать на дополнительные вопросы, в том числе непосредственно относящиеся к сути теоретического и практического вопросов билета.