

**Автономная некоммерческая организация профессионального образования  
«Колледж мировой экономики и передовых технологий»**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

учебной дисциплины

ОП. 03. Основы алгоритмизации и программирования

для специальности

**09.02.09 Веб-разработка**

квалификация – разработчик Веб-приложений

форма обучения – очная

Москва – 2025

## ОДОБРЕНА

Предметной (цикловой) комиссией  
математических дисциплин  
Протокол от 16 декабря 2025 г. № 4

**Разработана на основе Федерального  
государственного образовательного  
стандарта по специальности среднего  
профессионального образования 09.02.09 Веб-  
разработка**

Председатель ПЦК



/Космакова О.В.

Заместитель директора по методической работе

  
Подпись

/ Ю.И. Богомолова

## РАССМОТРЕНА

на заседании Педагогического совета  
Протокол от 18 декабря 2025 г. №3

Разработчик:

Космакова О.В., преподаватель АНО ПО «Колледж мировой экономики и передовых технологий»

## СОДЕРЖАНИЕ

<u>1.ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ учебной дисциплины ОП.04 Компьютерные сети</u> .....	4
2.КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ для текущего контроля успеваемости по дисциплине ОП. 03 Основы алгоритмизации и программирования .....	7
<u>2.КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ для промежуточного контроля успеваемости по дисциплине ОП. 03 Основы алгоритмизации и программирования</u> .....	10
3.1. Пояснительная записка.....	11
3.2. Оценочные средства промежуточного контроля.....	11

**1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ учебной дисциплины  
ОП. 03. Основы алгоритмизации и программирования.**

Результаты обучения (освоенные умения, усвоенные знания)	ПК, ОК	Наименование раздела, темы	Наименование оценочного средства	
			Текущий контроль	Промежуточная аттестация
<p>В результате освоения дисциплины обучающийся <i>должен знать</i>:</p> <p>понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;</p> <p>эволюцию языков программирования, их классификацию, понятие системы программирования;</p> <p>основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти; подпрограммы, составление библиотек подпрограмм.</p>	<p>ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5</p>	<p>Тема 1.1. Языки программирования</p>		
<p>В результате освоения дисциплины обучающийся <i>должен уметь</i>:</p> <p>разрабатывать алгоритмы для конкретных задач;</p> <p>использовать программы для графического отображения алгоритмов;</p> <p>определять сложность работы алгоритмов;</p> <p>работать в среде программирования;</p> <p>реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;</p> <p>оформлять код программы в соответствии со</p>	<p>ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5</p>	<p>Тема 1.2. Типы данных.</p>		

стандартом кодирования; выполнять проверку, отладку кода программы.	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5	Тема 2.1. Операторы языка программирования		
	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5	Тема 3.1. Процедуры и функции.		
	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2,	Тема 3.2. Структуризация в программировании		

	ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5			
	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5	<b>Тема 3.3.</b> Модульное программирование		
	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5	<b>Тема 4.1</b> Указатели		

	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5	<b>Тема 5.1</b> Основные принципы объектно- ориентированного программирования (ООП)		
	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5	<b>Тема 5.2</b> Интегрированная среда разработчика.		
	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2,	<b>Тема 5.3.</b> <b>Визуальное событийно-          управляемое          программирование</b>		

	ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5			
	ОК 1, ОК 2, ОК 4, ОК 5, ОК 9, ПК 1.1, ПК1.2, ПК 1.3, ПК 1.4, ПК 1.5, ПК 1.6, ПК 2.4, ПК 2.5	Дифференцированный зачет		

## 2. КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ

### для текущего контроля успеваемости по дисциплине ОП. 03. Основы алгоритмизации и программирования

#### 2.1. Пояснительная записка

Комплект оценочных средств предназначен для мониторинга качества получаемых обучающимися образовательных результатов, по наиболее значимым для дальнейшего обучения темам, разделам учебной дисциплины ОП. 03. Основы алгоритмизации и программирования и входит в состав фонда оценочных средств программы подготовки специалистов среднего звена по специальности 09.02.09 Веб-разработка, реализуемой в АНО ПО «Колледж мировой экономики и передовых технологий».

Комплект оценочных средств разработан в соответствии с рабочей программой ОП. 03. Основы алгоритмизации и программирования.

Комплект оценочные средства для проведения текущего контроля успеваемости включает:

- тесты, позволяющие провести процедуру измерения уровня знаний и умений обучающихся;
- опрос (устный/письменный), позволяющий оценить объем и глубину знаний по теме;
- контрольные материалы для проведения текущего контроля выполнения практических задач;
- проверка внеаудиторной самостоятельной работы, позволяющая оценить исполнительские навыки обучающихся в решении поставленных задач.

В результате освоения учебной дисциплины ОП. 03. Основы алгоритмизации и программирования обучающийся должен обладать предусмотренными 09.02.09 Веб-разработка умениями и знаниями:

Умения:	<ul style="list-style-type: none"><li>– разрабатывать алгоритмы для конкретных задач;</li><li>– использовать программы для графического отображения алгоритмов;</li><li>– определять сложность работы алгоритмов; - работать в среде программирования;</li><li>– реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;</li><li>– оформлять код программы в соответствии со стандартом кодирования;</li><li>– выполнять проверку, отладку кода программы.</li></ul>
Знания:	<ul style="list-style-type: none"><li>– понятие алгоритмизации, свойства алгоритмов, общие принципы построения алгоритмов, основные алгоритмические конструкции;</li><li>– эволюцию языков программирования, их классификацию, понятие системы программирования;</li><li>– основные элементы языка, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, классы памяти;</li><li>– подпрограммы, составление библиотек подпрограмм</li></ul>

Знания и умения, формируемые в рамках ОП. 03. Основы алгоритмизации и программирования, направлены на формирование общих и/или профессиональных компетенций:

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 04. Эффективно взаимодействовать и работать в коллективе и команде;

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

ОК 09. Пользоваться профессиональной документацией на государственном и иностранном языках.

ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием.

ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.

ПК 1.4. Выполнять тестирование программных модулей.

ПК 1.5. Осуществлять рефакторинг и оптимизацию программного кода.

ПК 1.6. Разрабатывать модули программного обеспечения для мобильных платформ.

ПК 2.4. Осуществлять разработку тестовых наборов и тестовых сценариев для программного обеспечения.

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

## **2.2. Оценочные средства для проведения тестирования**

1. Что такое алгоритм?

- а) Последовательность команд для выполнения задачи
- б) Программа на языке программирования
- в) Случайный набор инструкций
- г) Способ решения математической задачи

2. Какие основные свойства алгоритма существуют?

- а) Детерминированность, конечность, результативность, дискретность, понятность
- б) Скорость, стабильность, гибкость, масштабируемость
- в) Читаемость, надежность, безопасность, эффективность
- г) Простота, элегантность, универсальность, красота

3. Какой из перечисленных способов представления алгоритма является графическим?

- а) Псевдокод
  - б) Словесное описание
  - в) Блок-схема
  - г) Таблица
4. Что такое переменная в программировании?
- а) Константа, значение которой не может быть изменено
  - б) Именованная область памяти для хранения данных
  - в) Оператор, выполняющий определенное действие
  - г) Функция, возвращающая результат
5. Какой тип данных используется для хранения целых чисел?
- а) float (вещественное число)
  - б) string (строка)
  - в) int (целое число)
  - г) boolean (логическое значение)
6. Какой оператор используется для присваивания значения переменной?
- а) +
  - б) =
  - в) ==
  - г) >
7. Что такое цикл в алгоритмизации?
- а) Последовательное выполнение команд один раз
  - б) Повторяющееся выполнение блока команд
  - в) Условие, определяющее ход выполнения программы
  - г) Выделение памяти для данных
8. Какой тип цикла выполняет блок команд один раз, а затем проверяет условие выхода?
- а) Цикл for
  - б) Цикл while
  - в) Цикл do-while
  - г) Все вышеперечисленные
9. Что такое условный оператор (ветвление)?
- а) Блок команд, который выполняется всегда
  - б) Способ повторения блока команд
  - в) Инструкция, позволяющая выбрать выполнение одного из нескольких вариантов кода
  - г) Именованная область памяти
10. Какой условный оператор проверяет несколько условий последовательно?
- а) if
  - б) if-else
  - в) else if (или elif)
  - г) switch (или case)
11. Что такое функция (или подпрограмма)?
- а) Отдельная переменная

- б) Блок кода, выполняющий определенную задачу и который можно вызывать многократно
- в) Оператор сравнения
- г) Элемент пользовательского интерфейса

12. Как называется процесс определения порядка выполнения команд в программе?

- а) Тестирование
- б) Отладка
- в) Компиляция
- г) Алгоритмизация

13. Что такое компилятор?

- а) Программа, которая выполняет код пошагово
- б) Программа, переводящая исходный код с языка программирования в машинный код
- в) Инструмент для написания кода
- г) Система управления базами данных

14. Что такое отладка (debugging)?

- а) Процесс написания нового кода
- б) Процесс поиска и исправления ошибок в программе
- в) Оптимизация скорости выполнения программы
- г) Создание документации к программе

15. Какой оператор используется для вывода информации на экран?

- а) input()
- б) print()
- в) return()
- г) define()

16. Какое значение будет у переменной x после выполнения следующего кода (предположим, это Python)?

```
pythonКопировать
```

```
y = 10  
x = y * 2 + 5
```

- а) 15
  - б) 20
  - в) 25
  - г) 50
17. Что такое массив (или список)?
- а) Одна числовая переменная
  - б) Коллекция элементов одного или разных типов, доступных по индексу
  - в) Функция, возвращающая текст
  - г) Условный оператор

18. В каком случае оператор while будет бесконечным?

- а) Когда условие всегда истинно
- б) Когда условие всегда ложно
- в) Когда условие меняется

г) Когда внутри цикла нет никаких команд

19. Что такое синтаксическая ошибка?

а) Ошибка в логике программы

б) Нарушение правил написания кода на языке программирования

в) Проблема с внешним устройством

г) Неправильное использование математических формул

20. Какой из перечисленных языков программирования является высокоуровневым?

а) Ассемблер

б) Машинный код

в) C++

г) Машинный код и Ассемблер

Ответы к тестовым заданиям:

1. а

2. а

3. в

4. б

5. в

6. б

7. б

8. в

9. в

10.в

11.г

12.б

13.б

14.б

15.в

16.б

17.а

18.б

19.в

20.в