

МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Костромской государственный университет»  
(КГУ)

**ФОНД  
ОЦЕНОЧНЫХ СРЕДСТВ  
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ**

**ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ**

**Составлен в соответствии с учебным планом КГУ  
по программе подготовки специалистов среднего звена**

**по специальности  
09.02.10 Разработка компьютерных игр, дополненной  
и виртуальной реальности**

*Квалификация:* разработчик компьютерных игр, дополненной  
и виртуальной реальности

*Форма обучения очная*

**Кострома  
2025**

Разработал: Борисов А.С., и.о. директора Института «Высшая ИТ-школа»

УТВЕРЖДЕНО:

На заседании кафедры Прикладной математики и информатики, протокол № 3 от 17.12.2024 г.

# 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

## 1.1. Компетенции и индикаторы формируемые в процессе изучения дисциплины

Код компетенции	Содержание компетенции	Знания, умения (индикаторы компетенции)
ОК 01	Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	<p><b>Знания:</b></p> <p>31) актуальный профессиональный и социальный контекст, в котором приходится работать и жить;</p> <p>32) основные источники информации и ресурсы для решения задач и проблем в профессиональном и/или социальном контексте;</p> <p>33) алгоритмы выполнения работ в профессиональной и смежных областях;</p> <p>34) методы работы в профессиональной и смежных сферах; структуру плана для решения задач;</p> <p>35) порядок оценки результатов решения задач профессиональной деятельности.</p> <p><b>Умения:</b></p> <p>У1) распознавать задачу и/или проблему в профессиональном и/или социальном контексте;</p> <p>У2) анализировать задачу и/или проблему и выделять её составные части;</p> <p>У3) определять этапы решения задачи;</p> <p>У4) выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы;</p> <p>У5) составлять план действия; определять необходимые ресурсы; владеть актуальными методами работы в профессиональной и смежных сферах;</p> <p>У6) реализовывать составленный план;</p> <p>У7) оценивать результат и последствия своих действий (самостоятельно или с помощью наставника)</p>
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	<p><b>Знания:</b></p> <p>31) номенклатура информационных источников, применяемых в профессиональной деятельности;</p> <p>32) приемы структурирования информации;</p> <p>33) формат оформления результатов поиска информации, современные средства и устройства информатизации;</p> <p>34) порядок их применения и программное обеспечение в профессиональной деятельности, в том числе с использованием цифровых средств.</p>

		<p><b>Умения:</b></p> <p>У1) определять задачи для поиска информации; определять необходимые источники информации;</p> <p>У2) планировать процесс поиска; структурировать получаемую информацию;</p> <p>У3) выделять наиболее значимое в перечне информации; оценивать практическую значимость результатов поиска;</p> <p>У4) оформлять результаты поиска, применять средства информационных технологий для решения профессиональных задач;</p> <p>У5) использовать современное программное обеспечение; использовать различные цифровые средства для решения профессиональных задач</p>
ОК 05	Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	<p><b>Знания:</b></p> <p>31) особенности социального и культурного контекста;</p> <p>32) правила оформления документов и построения устных сообщений.</p> <p><b>Умения:</b></p> <p>У1) грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке, проявлять толерантность в рабочем коллективе</p>

## 1.2. Шкала оценивания сформированности компетенций

При оценивании сформированности компетенций по дисциплине «Основы алгоритмизации и программирования» используется 4-балльная шкала. Шкала соотносится с целями дисциплины и предполагаемыми результатами ее освоения.

Оценка «отлично» ставится, если обучающийся демонстрирует полное соответствие знаний, умений, навыков, приведенным в таблицах показателям, оперирует приобретенными знаниями, умениями, применяет их в ситуациях повышенной сложности.

Оценка «хорошо» ставится, если обучающийся демонстрирует соответствие знаний, умений, навыков, приведенным в таблицах показателям: основные знания, умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.

Оценка «удовлетворительно» ставится, если обучающийся демонстрирует неполное соответствие знаний, умений, навыков, приведенным в таблицах показателям: в ходе контрольных мероприятий обучающийся показывает владение менее 50% приведенных показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, обучающийся испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

Оценка «неудовлетворительно» ставится, если обучающийся демонстрирует полное отсутствие или явную недостаточность (менее 25%) знаний, умений, навыков в соответствие с приведенными показателями.

## 2. КОМПЛЕКТ КОНТРОЛЬНО-ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ (примерный)

### 2.1. Вопросы по темам/разделам дисциплины (примерные)

Контролируемый раздел дисциплины	Код контролируемой компетенции	Код индикатора компетенции	Примерный перечень вопросов/заданий для проверки сформированности индикаторов компетенций	Оценка уровня сформированности индикаторов
Раздел 1. Принципы построения алгоритмов и алгоритмические конструкции. Тема 1. Основные понятия алгоритмизации	ОК 01	33, У2, У5	Составьте словесное описание алгоритма проверки простоты числа, изобразите его в виде блок-схемы, составьте план действий для программной реализации.	Проверка правильности построения блок-схемы, обоснованность этапов алгоритма.
Раздел 1. Принципы построения алгоритмов и алгоритмические конструкции. Тема 2. Принципы построения и управляющие конструкции алгоритмического языка	ОК 02	34, У3, У6	Напишите код программы на любом языке программирования для расчёта факториала числа. Объясните использование управляющих конструкций (циклов) в решении задачи.	Проверка корректности кода, тестирование на нескольких входных данных.
Раздел 1. Принципы построения алгоритмов и алгоритмические конструкции. Тема 3. Логические основы алгоритмизации	ОК 01	34, У2, У7	Составьте логическое выражение для определения, попадает ли точка в область заданной фигуры (например, круга). Реализуйте алгоритм на выбранном языке программирования.	Демонстрация программы и устное пояснение логики условий.
Раздел 2. Системы и технологии структурного и объектно-ориентированного	ОК 02	33, У1, У4	Подготовьте сравнительную таблицу характеристик трёх современных IDE	Оценка полноты и обоснованности анализа.

программирования. Тема 4. Обзор современных систем программирования			(например, PyCharm, Visual Studio, Eclipse), указав их преимущества и недостатки для разных языков программирования.	
Раздел 2. Системы и технологии структурного и объектно-ориентированного программирования. Тема 5. Технология структурного программирования	ОК 01	33, У5, У6	Напишите программу для нахождения минимального элемента в массиве. Опишите структуру программы и объясните выбор используемого метода.	Проверка кода, оценка правильности результатов, устное объяснение выбора методов.
Раздел 2. Системы и технологии структурного и объектно-ориентированного программирования. Тема 6. Технология объектно-ориентированного программирования (ООП)	ОК 02	34, У4, У5	Разработайте класс "Книга" с атрибутами (название, автор, год издания) и методами (вывод информации, проверка актуальности). Протестируйте класс на нескольких объектах.	Проверка кода, тестирование работы методов класса, устное объяснение структуры класса.

## 2.2. Вопросы и задания к зачету

Структура зачета (примерная):

- Теоретический блок (30-40%): Вопросы, проверяющие знание основных понятий, принципов алгоритмизации и базовых элементов программирования.
- Практический блок (60-70%): Задания, требующие написания простых программ или фрагментов кода на одном из изучаемых языков программирования (например, Python, C#), а также разработки алгоритмов.
- Формат: Комбинированный (письменный ответ на теоретические вопросы и практическое выполнение заданий).
- Время: 2 - 2.5 часа (с учетом времени на написание кода).

Примерные вопросы для теоретического блока:

1. Основы алгоритмизации:

- Что такое алгоритм? Какие свойства алгоритма вы знаете?
- Какие способы записи алгоритмов вы знаете? (словесный, блок-схема, псевдокод)
- Какие основные типы алгоритмических структур существуют? (линейная, ветвление, цикл)
- Что такое блок-схема? Какие элементы она включает?

- Что такое псевдокод? Чем он отличается от языка программирования?
2. Основы программирования:
- Что такое переменная? Какие типы данных вы знаете?
  - Что такое оператор присваивания?
  - Какие основные арифметические и логические операции вы знаете?
  - Что такое условный оператор? (if, else)
  - Какие типы циклов вы знаете? (for, while)
  - Что такое массив? Как объявить и использовать массив?
  - Что такое функция? Зачем нужны функции?
  - Что такое область видимости переменной?
  - Что такое отладка программ? Какие методы отладки вы знаете?
3. Специфика разработки игр (основные понятия):
- Что такое игровой цикл?
  - Какие основные компоненты игры вы знаете? (графика, физика, ввод, звук)
  - Что такое спрайт?
  - Что такое коллизия?
  - Что такое вектор?

Примерные задания для практического блока:

1. Алгоритмы:

- Разработайте блок-схему алгоритма для вычисления факториала заданного числа.
- Запишите псевдокод для алгоритма поиска наибольшего элемента в массиве.
- Разработайте алгоритм для определения, является ли заданное число простым.

2. Программирование:

- Напишите программу на Python (или C#), которая выводит на экран таблицу умножения для заданного числа.
- Напишите программу, которая запрашивает у пользователя два числа и выводит на экран их сумму, разность, произведение и частное.
- Напишите программу, которая выводит на экран все четные числа из заданного массива.
- Напишите программу, которая вычисляет сумму элементов массива с помощью цикла for.
- Напишите функцию, которая вычисляет площадь треугольника по трем сторонам.
- Напишите функцию, которая принимает массив чисел и возвращает наибольшее из них.

3. Задачи, связанные с разработкой игр (базовые):

- Напишите фрагмент кода, который проверяет, произошло ли столкновение двух прямоугольных спрайтов.
- Напишите фрагмент кода, который перемещает спрайт на заданное расстояние по горизонтали.
- Определите вектор перемещения спрайта, если известны начальные и конечные координаты.
- Разработайте псевдокод для обработки ввода с клавиатуры (например, нажатие клавиш для перемещения персонажа).