**ДОКУМЕНТ ПОДПИСАН ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

**Сведения о сертификате ЭП**

Сертификат: **4187300118CCB3606C1AF1554B071263**
Владелец: **Бабин Владислав Николаевич**
Действителен: **с 08.12.2025 г. по 03.03.2027 г.**

ФГБОУ ВО Университет биотехнологий

Кафедра математики и физики

Рег. № ОПО 26-09

«22» 10 2026 г.

**УТВЕРЖДЕН**

на заседании кафедры

протокол от «22» 10 2026 г. № 3

Заведующий кафедрой

_____ Агафонова О.В.

# ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

## ОПЦ.03 Основы алгоритмизации и программирования

*Шифр и наименование дисциплины*

09.02.09 Веб-разработка

*Код и наименование направления подготовки*

Новосибирск 2026

## Паспорт фонда оценочных средств

| № п/п | Контролируемые разделы (темы) дисциплины* | Код контролируемой компетенции | Наименование оценочного средства |
|---|---|---|---|
| 1 | Раздел 1. Основы алгоритмизации<br>Тема 1.1. Алгоритм: понятие, свойства, способы описания | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 2 | Раздел 1. Основы алгоритмизации<br>Тема 1.2. Базовые алгоритмические конструкции: следование и ветвление | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 3 | Раздел 1. Основы алгоритмизации<br>Тема 1.3. Циклические алгоритмические конструкции | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 4 | Раздел 1. Основы алгоритмизации<br>Тема 1.4. Подпрограммы и вспомогательные алгоритмы | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 5 | Раздел 2. Структуры данных и типовые алгоритмы<br>Тема 2.1. Типы и структуры данных: массивы, записи, множества | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 6 | Раздел 2. Структуры данных и типовые алгоритмы<br>Тема 2.2. Алгоритмы поиска | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 7 | Раздел 2. Структуры данных и типовые алгоритмы<br>Тема 2.3. Алгоритмы сортировки | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 8 | Раздел 2. Структуры данных и типовые алгоритмы<br>Тема 2.4. Оценка вычислительной сложности алгоритмов | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 9 | Раздел 3. Основы технологии программирования<br>Тема 3.1. Этапы разработки программного средства. Трансляция и отладка | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 10 | Раздел 3. Основы технологии программирования<br>Тема 3.2. Структурное программирование | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 11 | Раздел 3. Основы технологии программирования<br>Тема 3.3. Основы объектно-ориентированного подхода | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 12 | Раздел 3. Основы технологии программирования<br>Тема 3.4. Стиль кодирования и документирование программ | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 13 | Раздел 4. Разработка и отладка алгоритмов<br>Тема 4.1. Тестирование и отладка алгоритмов | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 14 | Раздел 4. Разработка и отладка алгоритмов<br>Тема 4.2. Итоговая работа: разработка и обоснование алгоритма решения задачи | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Коллоквиум, собеседование; тестирование |
| 15 | Промежуточная аттестация: зачёт с оценкой (1 семестр) | ОК 01, ОК 02, ОК 05, ОК 09, ПК 1.1 | Вопросы к зачёту / зачёту с оценкой |

** Наименование разделов и тем берётся из рабочей программы дисциплины.*

## Матрица соответствия критериев оценки уровню сформированности компетенций

| Критерии оценки | Уровень сформированности компетенций |
|---|---|
| Отлично | Высокий уровень |
| Хорошо | Повышенный уровень |
| Удовлетворительно | Пороговый уровень |
| Неудовлетворительно | Недостаточный уровень |

**Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций**

1. Положение «О балльно-рейтинговой системе аттестации студентов» (http://edubiotech.ru/file/403: режим доступа свободный);

2. Положение «О проведении текущего контроля и промежуточной аттестации обучающихся» (http://edubiotech.ru/file/104821: режим доступа свободный).

## ТЕКУЩИЙ КОНТРОЛЬ УСПЕВАЕМОСТИ

**Вопросы для коллоквиумов и собеседования**

**Раздел 1. Основы алгоритмизации**

1. Дайте определение алгоритма и перечислите его свойства.
2. Сравните способы описания алгоритмов: словесный, блок-схема, псевдокод.
3. Перечислите основные условные обозначения блок-схем и назначение каждого.
4. Объясните алгоритмические конструкции «следование» и «ветвление».
5. Постройте блок-схему алгоритма нахождения наибольшего из трёх чисел.
6. Как строятся логические выражения в условиях ветвления? Приведите пример составного условия.
7. Что такое вложенные ветвления и в каких случаях их применяют?
8. Сравните циклы с предусловием, постусловием и параметром.
9. Составьте алгоритм суммирования последовательности чисел с помощью цикла с постусловием.
10. Что такое подпрограмма? Перечислите её параметры и результаты.
11. Опишите алгоритм и его свойства.
12. Опишите двоичный поиск и условие его применимости.
13. Раскройте понятия инкапсуляции, наследования и полиморфизма.
14. Объясните работу вложенных ветвлений на примере.

**Раздел 2. Структуры данных и типовые алгоритмы**

1. Чем отличаются статические и динамические структуры данных? Приведите примеры.
2. Охарактеризуйте массив: способ хранения, индексация, основные операции.
3. Что такое запись (структура) и в каких задачах её применяют?
4. Что такое множество как тип данных и какие операции над ним определены?
5. Опишите алгоритм линейного поиска и оцените его трудоёмкость.
6. Опишите алгоритм двоичного поиска. Каким требованиям должен удовлетворять массив?
7. Перечислите простые методы сортировки и сравните их по трудоёмкости.
8. Что такое устойчивость сортировки? Приведите пример неустойчивого метода.
9. Что означает оценка $O(f(n))$ и как сравнить $O(n)$ с $O(n \log n)$?
10. Как сравниваются алгоритмы по времени и по памяти? Приведите пример.
11. Объясните разницу между циклом с предусловием и циклом с постусловием.
12. Сравните простые методы сортировки по трудоёмкости.
13. Опишите линейный поиск и оцените его трудоёмкость.
14. Сравните асимптотические оценки $O(n)$ и $O(n^2)$.

**Раздел 3. Основы технологии программирования**

1. Перечислите этапы разработки программного средства.
2. Чем отличаются трансляция и интерпретация? Что такое отладка?
3. Что такое жизненный цикл программы?
4. Каковы принципы структурного программирования?
5. Что такое модульность, сцепление и связность модулей? Как они влияют на качество программы?
6. Что такое класс и объект? Приведите пример.
7. Раскройте понятия инкапсуляции, наследования и полиморфизма.
8. Что такое соглашения об именовании и зачем они нужны?
9. Какие требования предъявляются к комментариям в программе?
10. Что включает оформление программной документации?
11. Раскройте понятие подпрограммы и её параметров.
12. Перечислите этапы разработки программного средства.
13. Что такое устойчивость сортировки?
14. Перечислите требования к стилю кодирования и документированию программ.

**Раздел 4. Разработка и отладка алгоритмов**

1. Что такое тестовый набор и как его подготовить для алгоритма?
2. Что такое трассировка алгоритма и когда её применяют?
3. Чем отличаются пошаговая отладка и тестирование «чёрным ящиком»?
4. Какие граничные значения следует проверять при тестировании алгоритма?
5. Опишите этапы разработки и обоснования алгоритма решения задачи (итоговая работа).
6. Как выбрать структуру данных для задачи? Назовите критерии выбора.
7. Как оценить сложность итогового алгоритма и обосновать выбор?
8. Какие ошибки чаще всего допускают при разработке алгоритмов?
9. Как оформить обоснование алгоритма в итоговой работе?
10. Как проверить корректность алгоритма на вырожденных входных данных?
11. Охарактеризуйте массив и запись как структуры данных.
12. Что такое сцепление и связность модулей?
13. Сравните пошаговую отладку и тестирование «чёрным ящиком».

**Практические задания (умения)**

**Раздел 1. Основы алгоритмизации**

1. Составьте блок-схему алгоритма нахождения минимального числа из N введённых чисел (N задано).
2. Составьте трассировочную таблицу для алгоритма вычисления факториала числа N при N = 4.

**Раздел 2. Структуры данных и типовые алгоритмы**

1. Составьте псевдокод сортировки выбором для массива из 5 элементов и проследите её шаги.

**Раздел 3. Основы технологии программирования**

1. Составьте алгоритм подсчёта количества чётных элементов массива с оформлением подпрограммы.

**Раздел 4. Разработка и отладка алгоритмов**

1. Подготовьте не менее пяти тестовых наборов для алгоритма определения високосного года.

**Тестовые задания закрытого типа (выберите один правильный ответ)**

1. Какой термин означает конечную последовательность действий, приводящих к результату?

а) алгоритм
б) программа
в) данные
г) модуль

2. Какой блок блок-схемы обозначает условие (ветвление)?

а) прямоугольник
б) овал
в) параллелограмм
г) ромб

3. Какая конструкция выполняет действия, пока условие истинно, проверяя его перед выполнением?

а) цикл с постусловием
б) цикл с предусловием
в) ветвление
г) подпрограмма

4. Что такое подпрограмма?

а) блок данных
б) условие ветвления
в) именованная часть алгоритма, которую можно вызывать многократно
г) тип данных

5. Какой массив необходим для двоичного поиска?

а) любой неотсортированный
б) состоящий из одного элемента
в) упорядоченный
г) пустой

6. Какая трудоёмкость у линейного поиска в худшем случае?
   а) O(n)
   б) O(1)
   в) O(log n)
   г) O(n²)

7. Что такое устойчивая сортировка?
   а) сортировка без сравнений
   б) сортировка, работающая только с числами
   в) сортировка без памяти
   г) сортировка, сохраняющая порядок равных элементов

8. Какая трудоёмкость у пузырьковой сортировки в среднем?
   а) O(n)
   б) O(n²)
   в) O(n log n)
   г) O(log n)

9. Что такое статическая структура данных?
   а) структура, размер которой меняется во время работы
   б) файл на диске
   в) структура фиксированного размера, выделяемая заранее
   г) алгоритм

10. Что такое запись в алгоритме?
   а) цикл
   б) тип данных, объединяющий поля разных типов
   в) ветвление
   г) подпрограмма

11. Что такое модульность программы?
   а) разбиение программы на независимые части
   б) объединение всех функций в один блок
   в) запрет на подпрограммы
   г) использование только одного цикла

12. Что такое класс в объектно-ориентированном подходе?
   а) конкретный экземпляр
   б) тип сортировки
   в) файл
   г) шаблон, описывающий свойства и методы объектов

13. Что такое отладка программы?
   а) написание документации
   б) компиляция
   в) развёртывание на сервере

г) поиск и исправление ошибок

14. Что такое тест «чёрного ящика»?

а) проверка каждой строки кода
б) проверка функций без знания внутреннего кода
в) проверка производительности сети
г) проверка оформления

15. Какие значения входных данных важно проверять при тестировании?

а) случайные без анализа
б) только средние
в) граничные
г) только нулевые

16. Что такое трассировка алгоритма?

а) пошаговое отслеживание значений переменных
б) установка программы
в) подготовка отчёта
г) выбор языка программирования

*Ключ к тестам: 1–а, 2–г, 3–б, 4–в, 5–в, 6–а, 7–г, 8–б, 9–в, 10–б, 11–а, 12–г, 13–г, 14–б, 15–в, 16–а*

**Тестовые задания открытого типа (запишите ответ)**

1. Как называется свойство алгоритма, означающее, что он завершается за конечное число шагов?

2. Какую структуру данных работает по принципу LIFO?

3. Какова временная сложность бинарного поиска в отсортированном массиве?

4. Как называется алгоритм сортировки, который делит массив на части и сливает их?

5. Как называется ошибка, при которой программа выполняется, но дает неверный результат?

Ключ к открытым тестам: 1 – Конечность (результативность); 2 – Стек; 3 – O(log n); 4 – Сортировка слиянием (merge sort); 5 – Логическая ошибка.

**Критерии оценки текущего контроля (коллоквиум, собеседование)**

Отлично — Ответ полный, логичный, обоснованный. Обучающийся демонстрирует глубокое понимание материала, свободно отвечает на дополнительные вопросы, приводит примеры.

Хорошо — Ответ по существу верный, но содержит незначительные неточности. Обучающийся владеет материалом, но допускает несущественные ошибки при ответе на дополнительные вопросы.

Удовлетворительно — Ответ неполный, обучающийся демонстрирует знание основных положений, но затрудняется с пояснениями и примерами.

Неудовлетворительно — Ответ неверный или отсутствует. Обучающийся не усвоил основной материал, не может ответить на дополнительные вопросы.

## ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИЯ

Форма промежуточной аттестации: зачёт с оценкой (1 семестр).

**Вопросы для зачёта с оценкой**

1. Алгоритм: понятие, свойства, способы описания.
2. Блок-схемы: условные обозначения и правила построения.
3. Алгоритмические конструкции: следование и ветвление.
4. Циклические алгоритмические конструкции.
5. Подпрограммы и вспомогательные алгоритмы.
6. Типы и структуры данных: массивы, записи, множества.
7. Алгоритмы поиска: линейный и двоичный.
8. Алгоритмы сортировки и их устойчивость.
9. Оценка вычислительной сложности алгоритмов.
10. Этапы разработки программного средства. Трансляция и отладка.
11. Структурное программирование, модульность, сцепление и связность.
12. Основы объектно-ориентированного подхода.
13. Стиль кодирования и документирование программ.
14. Тестирование и отладка алгоритмов.
15. Разработка и обоснование алгоритма решения задачи (итоговая работа).

**Критерии оценки на зачёте с оценкой**

Отлично — Ответ полный, логичный, обоснованный. Обучающийся демонстрирует глубокое понимание материала, свободно отвечает на дополнительные вопросы, приводит примеры.

Хорошо — Ответ по существу верный, но содержит незначительные неточности. Обучающийся владеет материалом, но допускает несущественные ошибки при ответе на дополнительные вопросы.

Удовлетворительно — Ответ неполный, обучающийся демонстрирует знание основных положений, но затрудняется с пояснениями и примерами.

Неудовлетворительно — Ответ неверный или отсутствует. Обучающийся не усвоил основной материал, не может ответить на дополнительные вопросы.