

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
ПО ДИСЦИПЛИНЕ**
Алгоритмы и структуры данных

Код модуля
1156457(1)

Модуль
Алгоритмы и структуры данных

Екатеринбург

Оценочные материалы составлены автором(ами):

№ п/п	Фамилия, имя, отчество	Ученая степень, ученое звание	Должность	Подразделение
1	Косолюбов Дмитрий Александрович	кандидат физико-математических наук, без ученого звания	Доцент	алгебры и фундаментальной информатики

Согласовано:

Управление образовательных программ

Ю.Д. Маева

Авторы:

- Косолюбов Дмитрий Александрович, Доцент, алгебры и фундаментальной информатики

1. СТРУКТУРА И ОБЪЕМ ДИСЦИПЛИНЫ Алгоритмы и структуры данных

1.	Объем дисциплины в зачетных единицах	4	
2.	Виды аудиторных занятий	Лекции Практические/семинарские занятия	
3.	Промежуточная аттестация	Экзамен	
4.	Текущая аттестация	Контрольная работа	3
		Домашняя работа	3

2. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ (ИНДИКАТОРЫ) ПО ДИСЦИПЛИНЕ МОДУЛЯ Алгоритмы и структуры данных

Индикатор – это признак / сигнал/ маркер, который показывает, на каком уровне обучающийся должен освоить результаты обучения и их предъявление должно подтвердить факт освоения предметного содержания данной дисциплины, указанного в табл. 1.3 РПМ-РПД.

Таблица 1

Код и наименование компетенции	Планируемые результаты обучения (индикаторы)	Контрольно-оценочные средства для оценивания достижения результата обучения по дисциплине
1	2	3
ОПК-4 -Способен решать задачи профессиональной деятельности с использованием информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	З-1 - Формулировать представления о роли современных информационно-коммуникационных технологий для решения задач профессиональной деятельности У-1 - Обосновывать выбор современных ИТ-технологий для сбора, анализа, обработки и представления информации по профилю деятельности У-2 - Выбирать безопасные информационно-коммуникативные технологии для эффективного решения задач профессиональной деятельности	Домашняя работа № 1 Домашняя работа № 2 Домашняя работа № 3 Контрольная работа № 2 Контрольная работа № 3 Контрольная работа №1 Лекции Практические/семинарские занятия Экзамен

<p>ОПК-2 -Способен проводить под научным руководством исследования на основе современных методов в конкретной области профессиональной деятельности</p>	<p>Д-2 - Проявлять заинтересованность в содержании и результатах исследовательской работы З-1 - Демонстрировать понимание теоретических основ методов, используемых для проведения научных исследований в профильной области П-1 - Иметь опыт выполнения стандартных исследований с использованием серийного научного и технологического оборудования, стандартной методологии и методов исследований У-1 - Соотносить цель и задачи исследования с набором методов исследования, выбирать необходимое сочетание цели и средств</p>	<p>Домашняя работа № 1 Домашняя работа № 2 Домашняя работа № 3 Контрольная работа № 2 Контрольная работа № 3 Контрольная работа №1 Лекции Практические/семинарские занятия Экзамен</p>
<p>ПК-1 -Способен демонстрировать общенаучные базовые знания в математических и естественных науках, фундаментальной информатики и информационных технологиях</p>	<p>П-1 - Иметь практический опыт сбора информации в математических и естественных науках, фундаментальной информатики и информационных технологиях</p>	<p>Домашняя работа № 1 Домашняя работа № 2 Домашняя работа № 3 Контрольная работа № 2 Контрольная работа № 3 Контрольная работа №1 Лекции Практические/семинарские занятия Экзамен</p>

3. ПРОЦЕДУРЫ КОНТРОЛЯ И ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ В РАМКАХ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ МОДУЛЯ В БАЛЬНО-РЕЙТИНГОВОЙ СИСТЕМЕ (ТЕХНОЛОГИЧЕСКАЯ КАРТА БРС)

3.1. Процедуры текущей и промежуточной аттестации по дисциплине

1. Лекции: коэффициент значимости совокупных результатов лекционных занятий – 0.50		
Текущая аттестация на лекциях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
<i>домашняя работа</i>	1,4	20
<i>домашняя работа</i>	1,6	20
<i>домашняя работа</i>	1,12	20
<i>контрольная работа</i>	1,7	20

<i>контрольная работа</i>	1,16	20
Весовой коэффициент значимости результатов текущей аттестации по лекциям – 0.60		
Промежуточная аттестация по лекциям – экзамен		
Весовой коэффициент значимости результатов промежуточной аттестации по лекциям – 0.40		
2. Практические/семинарские занятия: коэффициент значимости совокупных результатов практических/семинарских занятий – 0.50		
Текущая аттестация на практических/семинарских занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
<i>Контрольная работа</i>	1,16	100
Весовой коэффициент значимости результатов текущей аттестации по практическим/семинарским занятиям– 1.00		
Промежуточная аттестация по практическим/семинарским занятиям– нет		
Весовой коэффициент значимости результатов промежуточной аттестации по практическим/семинарским занятиям– 0.00		
3. Лабораторные занятия: коэффициент значимости совокупных результатов лабораторных занятий –не предусмотрено		
Текущая аттестация на лабораторных занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент значимости результатов текущей аттестации по лабораторным занятиям -не предусмотрено		
Промежуточная аттестация по лабораторным занятиям – нет		
Весовой коэффициент значимости результатов промежуточной аттестации по лабораторным занятиям – не предусмотрено		
4. Онлайн-занятия: коэффициент значимости совокупных результатов онлайн-занятий –не предусмотрено		
Текущая аттестация на онлайн-занятиях	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент значимости результатов текущей аттестации по онлайн-занятиям -не предусмотрено		
Промежуточная аттестация по онлайн-занятиям – нет		
Весовой коэффициент значимости результатов промежуточной аттестации по онлайн-занятиям – не предусмотрено		

3.2. Процедуры текущей и промежуточной аттестации курсовой работы/проекта

Текущая аттестация выполнения курсовой работы/проекта	Сроки – семестр, учебная неделя	Максимальная оценка в баллах
Весовой коэффициент текущей аттестации выполнения курсовой работы/проекта– не предусмотрено		
Весовой коэффициент промежуточной аттестации выполнения курсовой работы/проекта– защиты – не предусмотрено		

4. КРИТЕРИИ И УРОВНИ ОЦЕНИВАНИЯ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ МОДУЛЯ

4.1. В рамках БРС применяются утвержденные на кафедре/институте критерии (признаки) оценивания достижений студентов по дисциплине модуля (табл. 4) в рамках контрольно-оценочных мероприятий на соответствие указанным в табл.1 результатам обучения (индикаторам).

Таблица 4

Критерии оценивания учебных достижений обучающихся

Результаты обучения	Критерии оценивания учебных достижений, обучающихся на соответствие результатам обучения/индикаторам
Знания	Студент демонстрирует знания и понимание в области изучения на уровне указанных индикаторов и необходимые для продолжения обучения и/или выполнения трудовых функций и действий, связанных с профессиональной деятельностью.
Умения	Студент может применять свои знания и понимание в контекстах, представленных в оценочных заданиях, демонстрирует освоение умений на уровне указанных индикаторов и необходимых для продолжения обучения и/или выполнения трудовых функций и действий, связанных с профессиональной деятельностью.
Опыт /владение	Студент демонстрирует опыт в области изучения на уровне указанных индикаторов.
Другие результаты	Студент демонстрирует ответственность в освоении результатов обучения на уровне запланированных индикаторов. Студент способен выносить суждения, делать оценки и формулировать выводы в области изучения. Студент может сообщать преподавателю и коллегам своего уровня собственное понимание и умения в области изучения.

4.2 Для оценивания уровня выполнения критериев (уровня достижений обучающихся при проведении контрольно-оценочных мероприятий по дисциплине модуля) используется универсальная шкала (табл. 5).

Таблица 5

Шкала оценивания достижения результатов обучения (индикаторов) по уровням

Характеристика уровней достижения результатов обучения (индикаторов)				
№ п/п	Содержание уровня выполнения критерия оценивания результатов обучения (выполненное оценочное задание)	Шкала оценивания		
		Традиционная характеристика уровня		Качественная характеристика уровня
1.	Результаты обучения (индикаторы) достигнуты в полном объеме, замечаний нет	Отлично (80-100 баллов)	Зачтено	Высокий (В)

2.	Результаты обучения (индикаторы) в целом достигнуты, имеются замечания, которые не требуют обязательного устранения	Хорошо (60-79 баллов)		Средний (С)
3.	Результаты обучения (индикаторы) достигнуты не в полной мере, есть замечания	Удовлетворительно (40-59 баллов)		Пороговый (П)
4.	Освоение результатов обучения не соответствует индикаторам, имеются существенные ошибки и замечания, требуется доработка	Неудовлетворительно (менее 40 баллов)	Не зачтено	Недостаточный (Н)
5.	Результат обучения не достигнут, задание не выполнено	Недостаточно свидетельств для оценивания		Нет результата

5. СОДЕРЖАНИЕ КОНТРОЛЬНО-ОЦЕНОЧНЫХ МЕРОПРИЯТИЙ ПО ДИСЦИПЛИНЕ МОДУЛЯ

5.1. Описание аудиторных контрольно-оценочных мероприятий по дисциплине модуля

5.1.1. Лекции

Самостоятельное изучение теоретического материала по темам/разделам лекций в соответствии с содержанием дисциплины (п. 1.2. РПД)

5.1.2. Практические/семинарские занятия

Примерный перечень тем

1. Полиномиальный хеш
2. Амортизация и неасимптотические оптимизации
3. Хеш таблица
4. Алгоритмы A* и Флойда—Уоршелла
5. Динамическое программирование
6. Суффиксный массив

Примерные задания

Хеширование строк

Когда для разных строк s и t выполняется $h(s) = h(t)$ – это *коллизия*

Можно ли сопоставить каждой строке s 32-битовое $h(s)$ без коллизий?

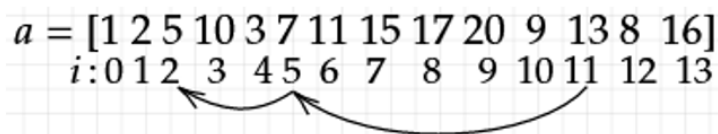
Нет: количество строк длины пять $\geq 2^{40}$ (пять байт), что больше чем 2^{32}



Пример: "1234", "1342", "2341", "4321" все имеют одинаковый хеш BadHashBySum

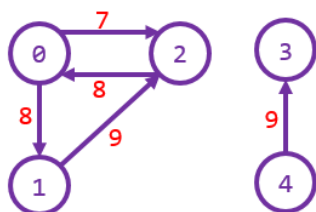
Бинарная куча

- полное бинарное дерево
- сыновья каждого узла имеют значения \geq значения родителя
- кучу из n элементов хранят в массиве $a[0..n-1]$
- $a[0]$ — корень
- сыновья $a[i]$ — $a[2i + 1]$ и $a[2i + 2]$
- родитель $a[i]$ при $i \neq 0$ — $a[\lfloor (i - 1)/2 \rfloor]$



Задачи

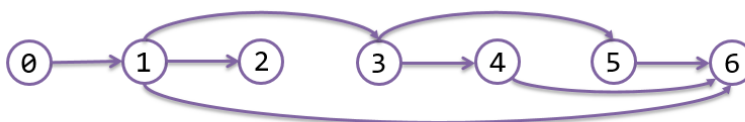
- Найти кратчайший путь из вершины s в вершину t в графе с неотрицательными весами на рёбрах
- Найти кратчайший путь из каждой вершины в каждую другую в графе с весами на рёбрах



Орграф (ориентированный граф) $G(V, E, c)$, где $E \subseteq V \times V$ и $c: E \rightarrow \mathbb{R}$ задаёт веса на рёбрах

Задача о числе путей

- орграф на n вершинах $0, 1, \dots, n-1$ и m рёбрах
- каждое ребро имеет вид (i, j) , где $i < j$
- найти количество путей из 0 в $n-1$
(полагая, что число-ответ влезет в int)



LMS-платформа – не предусмотрена

5.2. Описание внеаудиторных контрольно-оценочных мероприятий и средств текущего контроля по дисциплине модуля

Разноуровневое (дифференцированное) обучение.

Базовый

5.2.1. Контрольная работа №1

Примерный перечень тем

1. Алгоритм Карацубы

2. RAM-модель

Примерные задания

2. (Алгоритм Карацубы). Воспользуйтесь теперь следующей формулой для перемножения чисел x и y :

$$x \cdot y = 2^{64 \cdot n} a \cdot c + 2^{64 \cdot n/2} ((a + b)(c + d) - a \cdot c - b \cdot d) + b \cdot d$$

Заметьте, что для выполнения формулы достаточно всего лишь трёх умножений длинных чисел длиной $n/2$ с помощью рекурсии (формально $(a + b)(c + d)$ — это произведение чисел длиной $n/2 + 1$, но его легко свести к умножению двух чисел длиной $n/2$). Оцените время работы этого алгоритма. Оно должно быть $o(n^2)$.

3. В RAM модели размер машинного слова (int/ulong) не 64 бита, а $\lceil \log n \rceil$ битов или больше, где n — длина входа. Умножение и сложение $\lceil \log n \rceil$ -битовых чисел выполняется за $O(1)$. Пусть длинные числа x и y кодируются n битами: $x = \sum_{k=0}^{n-1} 2^k \cdot a_x[k]$ и $y = \sum_{k=0}^{n-1} 2^k \cdot a_y[k]$, где a_x и a_y — это массивы битов. Пользуясь возможностями RAM, ускорьте алгоритм Карацубы для умножения x и y в предположении, что размер машинного слова не превосходит $4 \lceil \log n \rceil$ (т.е. это word-RAM модель). Для простоты анализа считайте, что $n = 2^k \lceil \log n \rceil$ для некоторого $k \in \mathbb{N}$.

LMS-платформа – не предусмотрена

5.2.2. Контрольная работа № 2

Примерный перечень тем

1. Структура данных

Примерные задания

1. Рассмотрим следующую структуру данных: стек, в котором все элементы всегда отсортированы (самый верхний меньше всех), с операциями Pop — выкинуть самый верхний — и ForcePush(x) — выкинуть все элементы $\leq x$ и положить x на верх стека. Стек реализуется на динамическом массиве очевидным образом. Докажите, что амортизированное время работы операций Pop и ForcePush — $O(1)$.

2. Deque («дек», иногда пишут Dequeue) — это двусторонняя очередь с операциями PushFront(x), PopFront(), PushBack(x), PopBack(), которые вставляют/удаляют элемент в/из начала или конца очереди (у обычной очереди есть только PushBack и PopFront). Реализуйте дек с помощью массива так, чтобы амортизированное время работы всех операций было $O(1)$.

LMS-платформа – не предусмотрена

5.2.3. Контрольная работа № 3

Примерный перечень тем

1. Хеш таблица

Примерные задания

1. Дана строка $s[0..n-1]$. Рассмотрите хеш Карпа–Рабина: $h(t[0..m-1]) = \sum_{i=0}^{m-1} t[i] \cdot x^{m-i-1} \bmod p$, где p — это фиксированное простое число, такое что $p \geq \sigma$, где $\{0, 1, \dots, \sigma-1\}$ — это всевозможные значения символов s , и $x \in \{1, 2, \dots, p-1\}$ выбирается один раз случайно в самом начале работы программы.

- (a) Постройте структуру данных, которая отвечает с помощью хешей на запросы $\text{LCP}(i, j) = \max\{k : s[i..i+k-1] = s[j..j+k-1]\}$ (LCP от англ. Longest Common Prefix) за $O(\log n)$ времени. Оцените через параметры n и p вероятность неверного ответа на запрос. Каким должно быть p , чтобы при $n = 10\,000\,000$ вероятность ошибки на запрос была меньше $\frac{1}{10^9}$? Достаточно ли 64-битового числа для хранения такого p ?
- (b) Как отвечать на запросы за $O(\log \text{LCP}(i, j))$ времени примерно с той же вероятностью ошибки?

2. Рассмотрим структуру данных из лекции, которая на строке $s[0..n-1]$ с помощью хеша Карпа–Рабина отвечает на запросы $s[i..j] \stackrel{?}{=} s[i'..j']$ за $O(1)$. Из общей теории следует, что вероятность ошибки на запрос не больше $\frac{n-1}{p-1}$, что неплохо в решении большинства задач, особенно если p — это 64-битовое число $2^{64} - 59$ или $2^{61} - 1$ (получается $\frac{n-1}{p-1} < \frac{n}{2 \cdot 10^{18}}$). А какова вероятность, что построенная структура вообще безошибочная? Докажите, что она не меньше $1 - \frac{n^4}{p-1}$.

LMS-платформа – не предусмотрена

5.2.4. Домашняя работа № 1

Примерный перечень тем

1. Бинарное дерево

Примерные задания

1. Рассмотрим множество из n 4-байтовых чисел (тип `int`). Пусть адреса на нашей машине занимают 8 байтов. Тогда бинарное дерево поиска для этого множества занимает $24n$ байтов: 4 байта для каждого числа и по 8 байтов на указатели:

```
class Node { int x; Node* left, right; }
```

Зафиксируем $b = 16$. Каждый внутренний узел В-дерева, построенного на нашем множестве чисел, содержит от b до $2b$ элементов и занимает $4 + 8b + 8(2b + 1) = 24b + 12$ байтов: 4 байта на флаг `leaf` и счётчик `count` (+2 добавляется компилятором), $4 \cdot 2b$ байтов на массив из $2b$ элементов и по 8 байтов на каждый из $2b + 1$ указателей; каждый лист В-дерева занимает всего лишь $4 + 8b$ байтов:

```
class Leaf {
    bool leaf; //1 байтовый флаг
    int8 count; //1 байтовый счётчик
    int keys[2 * b];
};
class Node : Leaf {
    Node* kids[2 * b + 1];
};
```

(а) Сколько минимум/максимум листьев имеет дерево высоты h , в котором все листья на одном уровне и каждый нелистовой узел имеет от $b + 1$ до $2b + 1$ сыновей?

(б) Какова минимальная h_{\min} и максимальная h_{\max} высота В-дерева на n элементах? Для простоты считайте, что у корня есть как минимум $b + 1$ сын.

(в) Оцените, сколько памяти занимает В-дерево на n элементах при высоте h_{\min} и h_{\max} . Сравните результат с памятью в бинарном дереве поиска. Во сколько раз они отличаются?

LMS-платформа – не предусмотрена

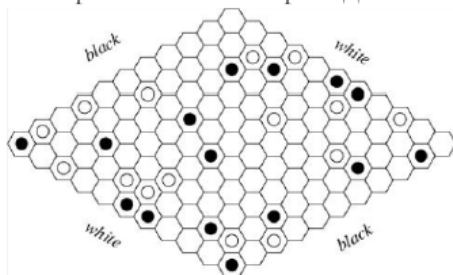
5.2.5. Домашняя работа № 2

Примерный перечень тем

1. Суффиксный массив

Примерные задания

1. Игра в камешки проходит на следующем (изначально пустом) поле из n клеток:



Два игрока ходят по-очереди. Чёрный игрок ставит чёрный камень на свободное поле, белый игрок — белый камень. Побеждает тот, кто соединил два “берега” своего цвета мостом из камней своего цвета. Вам дана последовательность из n ходов игроков и надо определить победный ход и кто победил за $o(n \log \log \log n)$ времени.

2. Покажите, что для СНМ без эвристики сжатия путей можно построить пример из m ($\geq n$) операций Union-Find, на котором время работы будет $\Omega(m \log n)$, где n — число элементов (таким образом, со сжатием путей действительно лучше, чем без него).

LMS-платформа – не предусмотрена

5.2.6. Домашняя работа № 3

Примерный перечень тем

1. Алгоритм поиска всех точек сочленения графа

Примерные задания

3. Дан взвешенный обыкновенный граф $G(V, E, c)$, где $c: E \rightarrow [0..∞)$ задаёт неотрицательные веса на рёбрах. С некоторыми вершинами v ассоциировано число $h_v \in \mathbb{N}$. Граф имитирует дорожную сеть с “блокпостами”. Каждая вершина $v \in V$ имеет один из трёх типов:

* Нейтральная — по ней можно ходить всегда,

* Дружественный блокпост — по ней можно ходить всегда и в этой вершине вам дают один жетон,

* Ограничивающий блокпост — через неё можно ходить, только если у вас с собой есть не менее h_v жетонов, полученных от дружественных блокпостов.

Изначально у вас ноль жетонов. Найти кратчайший путь из заданной вершины s в t за $O((n + m) \log n)$ времени. Важное ограничение: $h_v \leq 5$ (подсказка: константа 5 будет скрыта под O большим во времени работы).

LMS-платформа – не предусмотрена

5.3. Описание контрольно-оценочных мероприятий промежуточного контроля по дисциплине модуля

5.3.1. Экзамен

Список примерных вопросов

1. Модель RAM. Время работы в худшем случае. Ожидаемое время. Время работы быстрой сортировки. Лас-Вегас и Монте-Карло алгоритмы

2. Амортизированное время работы. Динамический массив: вставка и удаление. Библиотечные структуры данных, их время работы и память: динамические массивы, списки, стэки, очереди, словари, хеш таблицы; библиотечные алгоритмы. Память и кеш. Оптимизация доступа к кешу для умножения матриц и для кучи

3. Хеш таблица. Неизбежность коллизий. Выбор ячеек с помощью multiply-shift. Время работы вставок/удалений с multiply-shift. с-универсальные семейства хешей

4. Полиномиальный хеш. Неизбежность коллизий. Вероятностные гарантии отсутствия коллизий. Конкатенация и вычитание хешей. Сравнение подстрок за $O(1)$ хешами (Монте-Карло структура данных)

5. Дерево отрезков. Операция прибавления на отрезке и запрос суммы/минимума на отрезке

6. Персистентный массив с операциями изменения и чтения. Произвольные персистентные структуры данных на основе персистентного массива

7. B-дерево. Определение. Поиск, вставка и удаление. Время работы

8. Нагруженное B-дерево. Запросы суммы значений, ассоциированных с ключами из отрезка $[a, b]$. Время работы. Вставка/удаление в нагруженное B-дерево (без подробного описания алгоритма вставки/удаления)

9. Kope через B-дерево. Конкатенация верёвочных массивов. Время работы.

Нагруженный верёвочный массив (сумма/минимум элементов на отрезке массива)

10. Kope через B-дерево. Разбиение (split) верёвочных массивов. Время работы.

Персистентный верёвочный массив

11. Алгоритм поиска всех точек сочленения графа

12. Алгоритм поиска всех мостов графа

13. Алгоритм A^* . Примеры эвристик. Алгоритм A^* с бинарной кучей и его время работы. Неприменимость алгоритма A^* /Дейкстры для отрицательных весов. Алгоритм Флойда–Уоршелла
14. Системы непересекающихся множеств. Реализация с помощью деревьев. Время работы без сжатия путей. Итерированный логарифм. Определение псевдообратной функции Аккермана. Время работы со сжатием путей (без доказательства)
15. Динамическое программирование. Задача о количестве путей в ациклическом графе. Задача о минимальном гамильтоновом пути
16. Динамическое программирование. Задача о расписании: оптимизация используемой памяти и восстановление ответа. Задача о рюкзаке
17. Алгоритм построения Z-функции онлайн
18. Алгоритм построения сжатого бора для набора слов. Доказательство, что в сжатом боре всех подстрок будет не более $2n$ вершин. Поиск k слов длины m_1, m_2, \dots, m_k в строке длины n за $O(n \cdot \max\{m_1, \dots, m_k\} \cdot \log|\Sigma|)$, где Σ – это алфавит и $k < n$
19. Алгоритм построения суффиксного массива за $O(n \log n)$. Поразрядная сортировка. Структура Карпа–Миллера–Розенберга
20. Поиск подстроки в строке с помощью суффиксного массива. Обратный суффиксный массив. Алгоритм построения массива LCP. Вычисление наибольшего общего префикса двух подстрок строки
- LMS-платформа – не предусмотрена

5.4 Содержание контрольно-оценочных мероприятий по направлениям воспитательной деятельности

Направление воспитательной деятельности	Вид воспитательной деятельности	Технология воспитательной деятельности	Компетенция	Результаты обучения	Контрольно-оценочные мероприятия
Профессиональное воспитание	учебно-исследовательская, научно-исследовательская	Технология формирования уверенности и готовности к самостоятельной успешной профессиональной деятельности	ПК-1	П-1	Домашняя работа № 1 Домашняя работа № 2 Домашняя работа № 3 Контрольная работа № 2 Контрольная работа № 3 Контрольная работа № 1 Лекции Практические/семинарские занятия Экзамен