

**Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Московский физико-технический институт  
(национальный исследовательский университет)»**

**УТВЕРЖДЕНО**

**Директор физтех-школы  
радиотехники и компьютерных  
технологий**

**Д.А. Гаврилов**

**Рабочая программа дисциплины (модуля)**

<b>по дисциплине:</b>	Алгоритмы и структуры данных
<b>по направлению:</b>	Прикладные математика и физика
<b>профиль подготовки:</b>	Радиотехника и компьютерные технологии Физтех-школа Радиотехники и Компьютерных Технологий кафедра информатики и вычислительной техники
<b>курс:</b>	3
<b>квалификация:</b>	бакалавр

Семестр, формы промежуточной аттестации: 6 (весенний) - Дифференцированный зачет

Аудиторных часов: 30 всего, в том числе:

лекции: 30 час.

семинары: 0 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 15 час.

Всего часов: 45, всего зач. ед.: 1

Программу составил: Д.М. Альфонсо, старший преподаватель

Программа обсуждена на заседании кафедры информатики и вычислительной техники 13.01.2025

## Аннотация

Учебная дисциплина «Алгоритмы и структуры данных» знакомит студентов с фундаментальными понятиями, используемыми при разработке алгоритмов и оценке их качества. В учебную программу учебной дисциплины включены разделы, позволяющие строить эффективные алгоритмы для разнообразных задач дискретной и комбинаторной оптимизации с использованием различных структур данных.

Изучение учебной дисциплины позволяет дать студентам базу, необходимую для успешного усвоения материала перечисленных выше учебных дисциплин, а также получить знания необходимые им в дальнейшем для успешной работы.

### 1. Цели и задачи

#### Цель дисциплины

Цель преподавания учебной дисциплины «Алгоритмы и структуры данных» — формирование навыков для построения и анализа методов и алгоритмов при решении модельных задач дискретной оптимизации и их применение на практике. При изложении материала учебной дисциплины целесообразно выделить этап построения математической модели, существенно влияющей на ее адекватность реальной проблеме, а также показать возможность использования аппарата теории алгоритмов для анализа и обоснования выбора наиболее эффективных методов и алгоритмов для решения прикладных задач.

#### Задачи дисциплины

- формирование таких фундаментальных понятий, как размерность задачи и трудоемкость алгоритмов;
- изучение подходов для определения трудоемкости алгоритмов посредством составления и решения рекуррентных уравнений;
- изучение современных структур данных и обоснование выбора соответствующей структуры в зависимости от набора базовых операций, используемых в алгоритме.

### 2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
	УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
	УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи
ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
	ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации

ПК-4 Способен критически оценивать применимость используемых методик и методов	ПК-4.1 Знает численные порядки величин, характерных для соответствующей профессиональной области
	ПК-4.3 Способен обосновать причинно-следственные отношения используемых понятий и моделей

### 3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- основные виды структур данных, применяемых при решении задач;
- алгоритмы обработки информации, хранящейся в различных видах структур данных;
- достоинства и недостатки каждого вида структур данных для применения при решении различных задач.

уметь:

- делать обоснованный выбор используемых при решении задач структур данных;
- применять структуры данных и алгоритмы их обработки при решении различных задач.

владеть:

- навыками создания собственных и использования библиотечных структур данных при разработке программ на языке программирования C.

### 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

#### 4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

№	Тема (раздел) дисциплины	Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.			
		Лекции	Семинары	Лаборат. работы	Самост. работа
1	Базовые алгоритмы решения задач	5			2
2	Динамические структуры данных	5			1
3	Граф как структура данных	5			2
4	Деревья как частный случай графов	5			2
5	Деревья сортировки и сбалансированные деревья	10			8
Итого часов		30			15
Подготовка к экзамену		0 час.			
Общая трудоёмкость		45 час., 1 зач.ед.			

#### 4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 6 (Весенний)

##### 1. Базовые алгоритмы решения задач

Алгоритмы поиска. Алгоритмы сортировки: внутренняя и внешняя сортировка. Использование хеширования для поиска данных. Решение задач с перебором: алгоритмы типа "разделяй и властвуй", метод динамического программирования, метод ветвей и границ. Решение задач на тему "Сортировка и поиск" и "Метод ветвей и границ и динамическое программирование"

##### 2. Динамические структуры данных

Определение структуры данных. Линейные связанные списки: однонаправленные и двунаправленные. Очередь, стек, дек - реализации в виде массива и списка. Примеры приложений, использующих списки, стеки и очереди. Решение задач на темы использования динамических структур данных, особенно стеков и очередей.

### 3. Граф как структура данных

Граф как структура данных. Основные определения теории графов. Приложения, использующие графы как структуры данных. Представления графов в программах. Алгоритмы обхода графов : поиск в глубину и поиск в ширину. Алгоритмы поиска кратчайших путей: алгоритм Флойда и алгоритм Дейкстры. Построение кратчайших остовов : алгоритм Краскала. Решение задач на тему "Графы". Хранение графа в программе, проверки различных условий (полнота, связность, двудольность), поиск кратчайших путей и остовов минимального веса.

### 4. Деревья как частный случай графов

Определения ориентированного, упорядоченного, бинарного дерева. Представление деревьев в программе. Код Прюфера. Представление упорядоченных ориентированных деревьев. Представление бинарных деревьев. Решение задач на тему "Деревья". Использование деревьев для поиска.

### 5. Деревья сортировки и сбалансированные деревья

Определение дерева сортировки, приложения использования. Алгоритм поиска в дереве сортировки. Алгоритм вставки в дерево сортировки. Алгоритм удаления из дерева сортировки. Определение сбалансированного дерева. Балансировка деревьев. Решение задач на тему "Деревья сортировки". Операции с деревьями сортировки, использование. Операции по балансировке деревьев.

## 5. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Учебная аудитория, оснащенная компьютером и мультимедийным оборудованием (проектор, звуковая система).

## 6.Перечень рекомендуемой литературы

### Основная литература

1. Андрианова, А.А. Практикум по курсу "Алгоритмизация и программирование". Часть 1 с./А.А.Андрианова, Т.М.Мухтарова. - Казань: Казанский государственный университет, 2008.- 96
2. Андрианова, А.А. Практикум по курсу "Алгоритмизация и программирование". Часть 2. /А.А.Андрианова, Л.Н.Исмагилов, Т.М.Мухтарова. - Казань: Казанский государственный университет, 2009.- 132 с.
3. Долгов, А. И. Алгоритмизация прикладных задач [Электронный ресурс] : Уч. пособ / А. И. Долгов. - М. : Флинта, 2011. - 136 с. - ISBN 978-5-9765-0086-2.  
<http://www.znaniy.com/bookread.php?book=406093>

### Дополнительная литература

1. Алгоритмы и структуры данных. Новая версия для Оберона / Никлаус Вирт ; пер. с англ. Под ред. д.ф.-м.н. Ткачева Ф. В. ? Москва : ДМК Пресс, 2014 .? 272 . : ил. ; 21 .? (Классика программирования) .? 1-е изд. 2001 .? Библиогр. в конце гл. ? Предм. указ. : с. 270-272 .? ISBN 978-5-97060-011-5 ((в обл.)) , 200.
2. [Сборник задач по курсу "Алгоритмы и структуры данных"] [Текст: электронный ресурс] : практикум / А. А. Андрианова, Т. М. Мухтарова ; Казан. (Приволж.) федер. ун-т, Ин-т вычисл. математики и информац. технологий <URL:[http://libweb.ksu.ru/ebooks/09\\_63\\_ds021.pdf](http://libweb.ksu.ru/ebooks/09_63_ds021.pdf)>.
3. Гаврилова, И. В. Разработка приложений [Электронный ресурс] : учеб. пособие / И. В. Гаврилова., 2-е изд., стер., М.: ФЛИНТА, 2012, 242 с. - ISBN 978-5-9765-1482-9 <http://znanium.com/bookread.php?book=455037>
4. Структуры и алгоритмы обработки данных: Учебное пособие / В.Д. Колдаев. - М.: ИЦ РИОР:НИЦ ИНФРА-М, 2014. - 296 с.: 60х90 1/16. - (Высшее образование: Бакалавриат). ISBN 978-5-369-01264-2, 500 экз. <http://znanium.com/bookread.php?book=418290>
5. Кубенский А.А. Структуры и алгоритмы обработки данных: объектно-ориентированный подход и реализация на C++. - СПб.: БХВ-Петербург, 2004. - 464 с. -ISBN 5-94157-506-8. <http://znanium.com/bookread.php?book=356880>

## **7. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)**

[http://libweb.ksu.ru/ebooks/09\\_63\\_ds021.pdf](http://libweb.ksu.ru/ebooks/09_63_ds021.pdf)  
<http://znanium.com/bookread.php?book=455037>  
<http://znanium.com/bookread.php?book=356880>  
<http://znanium.com/bookread.php?book=418290>  
[www.znanium.com/bookread.php?book=406093](http://www.znanium.com/bookread.php?book=406093)

## **8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)**

На занятиях используются мультимедийные технологии, включая демонстрацию презентаций.

## **9. Методические указания для обучающихся по освоению дисциплины (модуля)**

Изучение дисциплины осуществляется по следующим формам: лекции и самостоятельная работа студента. Важным условием для освоения дисциплины в процессе занятий является ведение конспектов, освоение и осмысление терминологии изучаемой дисциплины. Материалы лекционных занятий следует своевременно подкреплять проработкой соответствующих разделов в учебниках, учебных пособиях, научных статьях и монографиях, в соответствии со списком основной и дополнительной литературы. В рамках изучения учебной дисциплины необходимо использовать передовые информационные технологии – компьютерную технику, электронные базы данных, Интернет.

Целями самостоятельной работы студента является:

- систематизация и закрепление полученных теоретических знаний и практических умений студентов;
- углубление и расширение теоретических знаний;
- формирование умения использовать справочную литературу;
- формирование самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;
- развитие исследовательских умений.

Самостоятельная работа выполняется студентом по заданию преподавателя и может содержать в себе следующее задания:

- изучение программного материала дисциплины (работа с учебником и конспектом лекции, изучение рекомендуемых литературных источников, конспектирование источников);
- решение задач;
- ответы на контрольные вопросы;
- выполнение контрольных работ;

- работа с электронными информационными ресурсами и ресурсами Internet (использование аудио- и видеозаписи);
- составление схем, таблиц для систематизации учебного материала;
- написание докладов, эссе, стихов, рефератов, конспектов занятий;
- аннотирование, реферирование, рецензирование текста;
- работа с компьютерными программами;
- выполнение тестовых заданий;
- подготовка презентаций;
- подготовка к дифференцированному зачету.

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

<b>по направлению:</b>	Прикладные математика и физика
<b>профиль подготовки:</b>	Радиотехника и компьютерные технологии Физтех-школа Радиотехники и Компьютерных Технологий кафедра информатики и вычислительной техники
<b>курс:</b>	<u>3</u>
<b>квалификация:</b>	бакалавр
Семестр, формы промежуточной аттестации: 6 (весенний) - Дифференцированный зачет	
<b>Разработчик:</b>	Д.М. Альфонсо, старший преподаватель

## 1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Анализирует задачу, выделяя этапы ее решения, действия по решению задачи
	УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи
	УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и недостатки
	УК-1.4 Грамотно, логично, аргументированно формирует собственные суждения и оценки
	УК-1.5 Определяет и оценивает практические последствия возможных вариантов решения задачи
ОПК-1 Способен применять фундаментальные знания, полученные в области физико-математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК-1.1 Способен анализировать поставленную задачу, намечать пути ее решения
	ОПК-1.2 Способен строить математические модели, производить количественные расчеты и оценки
	ОПК-1.3 Способен определять границы применимости полученных результатов
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации
ПК-4 Способен критически оценивать применимость используемых методик и методов	ПК-4.1 Знает численные порядки величин, характерных для соответствующей профессиональной области
	ПК-4.3 Способен обосновать причинно-следственные отношения используемых понятий и моделей

## 2. Показатели оценивания компетенций

В результате изучения дисциплины «Алгоритмы и структуры данных» обучающийся должен:

### знать:

- основные виды структур данных, применяемых при решении задач;
- алгоритмы обработки информации, хранящейся в различных видах структур данных;
- достоинства и недостатки каждого вида структур данных для применения при решении различных задач.

### уметь:

- делать обоснованный выбор используемых при решении задач структур данных;
- применять структуры данных и алгоритмы их обработки при решении различных задач.

### владеть:

- навыками создания собственных и использования библиотечных структур данных при разработке программ на языке программирования С.

## 3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

Текущий контроль осуществляется посредством устного опроса и контрольных работ в конце занятия по пройденной теме.

Примерные варианты вопросов для текущего контроля:

1. Определение класса
2. Виды членов класса



3. Области видимости членов класса
4. Операция доступа к членам класса
5. Статические члены класса
6. Встроенные классы int, double, char, byte
7. Класс Console
8. Класс Math
9. Возможности класса string
10. Массивы и возможности класса array
11. Виды конструкторов класса
12. Правила определения свойств класса
13. Использование индексаторов
14. Интерфейсы
15. Правила перегрузки операций
16. Наследование
17. Базовые и производные классы
18. Конструкторы производных классов
19. Определение делегатов
20. Вызов методов с помощью делегатов
21. Запись в файл объектов типа byte
22. Запись в файл чисел
23. Запись в файл текста
24. Сериализация и десериализация

#### **4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся**

1. Даны множества A и B из nA и nB целых положительных чисел (каждое число входит в соответствующее множество только один раз). Создать функцию TestInclu() для проверки принадлежности множества A множеству B. Функция возвращает 1, если A содержится в B, и 0 в противном случае.

Пример: A={3,4,1,6,7}, nA=5, B={8,1,3,10}, nB=4. Результат: TestInclu() возвращает 0.

2. Даны множества A и B из nA и nB целых положительных чисел (каждое число входит в соответствующее множество только один раз). Создать функцию Union(), которая строит их объединение A+B и возвращает адрес массива.

3. Даны множества A и B из nA и nB целых положительных чисел (каждое число входит в соответствующее множество только один раз). Создать функцию с именем Intersection для получения их пересечения A\*B (в A\*B не должно быть повторяющихся чисел).

Пример. A={3,4,1,6,7}, nA=5, B={8,1,3,10}, nB=4. Пересечение A\*B={3,1}, nA\*B=2.

4. Даны множества A и B из nA и nB целых положительных чисел (каждое число входит в соответствующее множество только один раз). Создать функцию Difference() для получения их разности A\B (в A\B не должно быть повторяющихся чисел).

Пример: A={3,4,1,6,7}, nA=5, B={8,1,3,10}, nB=4. Разность A\B={4,6,7}.

5. Дана последовательность A=(aj) 1≤j≤n из n целых чисел. Создать функцию с именем Count, возвращающую количество чисел, которые встречаются в A только один раз.

Пример: A={0,3,8,3,1,9,0,1,1}. Ответ: 2.

6. Для хранения многочлена P(x)=a0+a1x+...+anxn степени n используется массив. Определить функцию AddPolynom() для сложения двух многочленов.

7. Пусть A=(aj) 1≤j≤n - последовательность из n целых чисел. Число ak, 1≤k≤n, назовем локальным максимумом, если ak-1<ak>ak+1. Определить функцию IndexMax(), вычисляющую номер наибольшего локального максимума в A. При отсутствии локальных максимумов функция должна вернуть -1.

Пример. A={0,4,3,2,2,8,7}. Ответ: 5 (нумерация начинается с нуля).

8. (Слияние последовательностей) Пусть  $A=(a_i)_{1 \leq i \leq nA}$  и  $B=(b_i)_{1 \leq i \leq nB}$  – неубывающие последовательности из  $nA$  и  $nB$  целых чисел соответственно. Создать функцию `Merger()`, осуществляющую построение неубывающей последовательности  $C$  длины  $nA+nB$  из членов  $A$  и  $B$  и возвращающую адрес массива, в котором хранится  $C$ . В главной функции создать массивы  $s$   $A$  и  $B$ , получить с помощью `Merger()` массив  $C$  и отобразить его на дисплее.

Пример.  $A=\{1,4,5,6\}$ ,  $nA=4$ ,  $B=\{3,5,7\}$ ,  $nB=3$ .  $C=\{1,3,4,5,5,6,7\}$ ,  $nC=7$ .

9. (Неубывающие цепочки) Пусть  $A=(a_j)_{1 \leq j \leq n}$  – последовательность из  $n$  положительных целых чисел. Ее подпоследовательность  $a_{m+1}, a_{m+2}, \dots, a_{m+k}$  называется неубывающей цепочкой длины  $k$ , если  $a_{m+1} \leq a_{m+2} \leq \dots \leq a_{m+k}$  и  $a_m > a_{m+1}$ ,  $a_{m+k} > a_{m+k+1}$ . При этом  $a_1$  всегда является началом цепочки, а  $a_n$  – окончанием или началом. Определить функцию `CountSubSet()`, получающую вектор  $B=(b_j)_{1 \leq j \leq n}$ , у которого  $b_j$  является числом неубывающих цепочек длины  $j$ .

Пример:  $A=\{10, 3,3,15, 14,21\}$ ,  $n=6$ .  $B=\{1,1,1,0,0,0\}$ .

10. Задано вещественно (типа `double`) число  $x$  и полное имя двоичного файла, в котором хранится таблица вещественных чисел из  $h$  строк и  $w$  столбцов, записанная с помощью стандартной функции `fwrite()`. Определить функцию с именем `ReplaceItem()`, выполняющую замену в файле элемента из последней строки и последнего столбца числом  $x$ .

11. Дано полное имя файла, в котором хранятся векторы разной длины с вещественными (типа `double`) координатами, записанные с помощью стандартной функции `fprintf()`. При записи вектора вначале записывается его длина (тип `int`), а затем его вещественные координаты. Определить функцию `AddVector()`, которая записывает в конец данного файла вектор, длина и координаты которого вводятся с клавиатуры.

12. Дано полное имя файла, в котором хранится таблица вещественных (типа `double`) чисел, записанная с помощью стандартной функции `fprintf()`. Вначале записана ее высота  $h$  и ширина  $w$  (тип `int`), а затем – элементы. Определить функцию `GetColumn()`, которая читает из файла столбец с указанным номером  $j$  и сохраняет его в массиве.

13. Задан полный путь к файлу с таблицей вещественных (типа `double`) чисел, записанной с помощью стандартной функции `fwrite()`. Вначале записана ее высота  $h$  и ширина  $w$  (тип `int`), а затем – элементы. Определить функцию `Invert()`, которая получает файл с таблицей, транспонирует таблицу и записывает ее в другой файл.

14. Текст – последовательность символов, состоящая из букв и пробелов. Слово – подпоследовательность, не содержащая пробелов. Тексты и слова хранятся в символьных строках. Создать функцию `IndexWord()`, обеспечивающую проверку присутствия в тексте заданного слова и вычисляющую позицию (от начала текста) первой буквы первого вхождения этого слова. При отсутствии слова в тексте функция возвращает `-1`.

15. Написать собственный вариант `MyStrCmp()` стандартной функции `strcmp()`.

## Критерии оценивания

Оценка отлично (10) выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины, проявляющему интерес к данной предметной области, продемонстрировавшему умение уверенно и творчески применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично (9) выставляется студенту, показавшему всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений.

Оценка отлично (8) выставляется студенту, показавшему систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, правильное обоснование принятых решений, с некоторыми недочетами.

Оценка хорошо (7) выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но недостаточно грамотно обосновывает полученные результаты.

Оценка хорошо (6) выставляется студенту, если он твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности.

Оценка хорошо (5) выставляется студенту, если он в основном знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач достаточно большое количество неточностей.

Оценка удовлетворительно (4) выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он освоил основные разделы учебной программы, необходимые для дальнейшего обучения, и может применять полученные знания по образцу в стандартной ситуации.

Оценка удовлетворительно (3) выставляется студенту, показавшему фрагментарный, разрозненный характер знаний, допускающему ошибки в формулировках базовых понятий, нарушения логической последовательности в изложении программного материала, слабо владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и с трудом применяет полученные знания даже в стандартной ситуации.

Оценка неудовлетворительно (2) выставляется студенту, который не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных принципов и не умеет использовать полученные знания при решении типовых задач.

Оценка неудовлетворительно (1) выставляется студенту, который не знает основного содержания учебной программы дисциплины, допускает грубейшие ошибки в формулировках базовых понятий дисциплины и вообще не имеет навыков решения типовых практических задач.

## **5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности**

Дифференцированный зачет проводится в устной форме.

При проведении зачета обучающемуся предоставляется 20 минут на подготовку. Опрос обучающегося проводится в течение 40 минут.

Во время проведения дифференцированного зачета обучающиеся могут пользоваться программой дисциплины.