

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Луковникова Елена Ивановна  
Должность: Проректор по учебной работе  
Дата подписания: 16.11.2021 10:50:13  
Уникальный программный ключ:  
890f5aae3463de1924cbcf76ac5d7ab89e9fe7d3

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ  
ВЫСШЕГО ОБРАЗОВАНИЯ

"БРАТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ"



УТВЕРЖДАЮ

Проректор по учебной работе

*Солухов*  
Е.И.Луковникова

*31.11.2021*  
20 21 г.

## РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

### Б1.О.02 Алгоритмы и структуры данных

Закреплена за кафедрой **Информатики, математики и физики**

Учебный план b010302\_21\_ИПО.plx

Направление: 01.03.02 Прикладная математика и информатика

Квалификация **Бакалавр**

Форма обучения **очная**

Общая трудоемкость **9 ЗЕТ**

Виды контроля в семестрах:

Экзамен 1,2, Курсовая работа 2

### Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	1 (1.1)		2 (1.2)		Итого	
	17	18	17	18		
Неделя	17	18				
Вид занятий	УП	РП	УП	РП	УП	РП
Лекции	17	17	36	36	53	53
Лабораторные			36	36	36	36
Практические	34	34			34	34
В том числе инт.	12	12	16	16	28	28
Итого ауд.	51	51	72	72	123	123
Контактная работа	51	51	72	72	123	123
Сам. работа	39	39	72	72	111	111
Часы на контроль	54	54	36	36	90	90
Итого	144	144	180	180	324	324

Программу составил(и):

к.т.н., доц., Сташок О.В.; б.с., ст.пр., Ратинская Е.В.

Сташок Е.В.

Рабочая программа дисциплины

### Алгоритмы и структуры данных

разработана в соответствии с ФГОС:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 01.03.02 Прикладная математика и информатика (приказ Минобрнауки России от 10.01.2018 г. № 9)

составлена на основании учебного плана:

Направление: 01.03.02 Прикладная математика и информатика  
утвержденного приказом ректора от 01.03.2021 протокол № 80.

Рабочая программа одобрена на заседании кафедры

### Информатики, математики и физики

Протокол от 16 04 2024 г. № 9

Срок действия программы: 2021 - 2025 уч.г.

Зав. кафедрой Горохов Д.Б.

Д.Б. Горохов

Председатель МКФ

старший преподаватель Латушкина С.В.

18 до апреля

2024 г.

С.В. Латушкина

Ответственный за реализацию ОПОП

Д.Б. Горохов  
(подпись)

Горохов Д.Б.  
(ФИО)

Директор библиотеки

Соболева  
(подпись)

Соболева Л.П.  
(ФИО)

№ регистрации

03  
(методический отдел)

**1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

1.1	ознакомление обучающихся с математическими моделями алгоритмов, основами теории сложности вычислений, привить практические навыки конструирования алгоритмов решения распространенных математических задач.
-----	---

**2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП**

Цикл (раздел) ООП:	Б1.О.02
<b>2.1</b>	<b>Требования к предварительной подготовке обучающегося:</b>
2.1.1	Для успешного освоения дисциплины студент должен иметь базовую подготовку по элементарной математике в объеме программы средней школы.
<b>2.2</b>	<b>Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:</b>
2.2.1	Языки и методы программирования
2.2.2	Учебная практика (проектно-технологическая)

**3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)**

**УК-2: Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений**

Индикатор 1	УК-2.1 Формулирует в рамках поставленной цели проекта совокупность задач, обеспечивающих ее достижение
<b>ОПК-2: Способен использовать и адаптировать существующие математические методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач</b>	
Индикатор 1	ОПК-2.1 Использует существующие методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач
Индикатор 2	ОПК-2.2 Адаптирует существующие методы и системы программирования для разработки и реализации алгоритмов решения прикладных задач

**В результате освоения дисциплины обучающийся должен**

<b>3.1</b>	<b>Знать:</b>
3.1.1	способы достижения результатов в рамках поставленной цели; основы алгоритмизации и программирования; приемы адаптации методов и систем программирования в решении прикладных задач.
<b>3.2</b>	<b>Уметь:</b>
3.2.1	проводить анализ поставленной цели и формулировать задачи, необходимые для ее достижения; анализировать альтернативные варианты; составлять алгоритмы решения задач различной структуры, применять языки программирования; адаптировать методы и системы программирования к разработке и реализации алгоритмов решения прикладных задач.
<b>3.3</b>	<b>Владеть:</b>
3.3.1	методиками разработки цели и задач проекта; навыками использования существующих методов и систем программирования для разработки и реализации алгоритмов решения прикладных задач; навыками использования существующих методов и систем программирования для адаптации алгоритмов к решению прикладных задач.

**4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

Код занятия	Вид занятия	Наименование разделов и тем	Семестр / Курс	Часов	Компетенции	Литература	Инте ракт.	Примечание
	Раздел	<b>Раздел 1. Теоретические основы теории алгоритмов</b>						
1.1	Лек	Основные понятия теории алгоритмов	1	1	УК-2 ОПК-2	Л1.2 Л1.3Л3.1	0	УК-2.1,ОПК-2.2
1.2	Лек	Математические модели алгоритмов	1	4	УК-2	Л1.2 Л1.3Л2.2	2	лекция-беседа УК-2.1
1.3	Лек	Теоретико-числовые алгоритмы	1	2	УК-2	Л1.2 Л1.3Л2.3 Э1 Э2 Э3	2	лекция-беседа УК-2.1
1.4	Лек	Сложность алгоритмов	1	6	УК-2	Л1.2 Л1.3Л2.2 Э3	0	УК-2.1
1.5	Лек	Приближенные и вероятностные алгоритмы	1	4	УК-2	Л1.2 Л1.3Л2.2 Э2 Э3	0	УК-2.1

1.6	Пр	Математические модели алгоритмов	1	8	ОПК-2	Л1.2Л3.1 Э2 Э3	0	ОПК-2.1
1.7	Пр	Основы языка Python	1	8	ОПК-2	Л1.1 Л1.2 Э3	4	работа в малых группах ОПК-2.1
1.8	Пр	Анализ сложности алгоритмов	1	8	УК-2 ОПК-2	Л1.2Л2.2Л3.1 Э3	0	УК-2.1,ОПК-2.2
1.9	Пр	P и NP сложные задачи	1	6	ОПК-2	Л1.1 Л1.2Л3.1 Э3	0	ОПК-2.1,ОПК-2.2
1.10	Пр	Жадные алгоритмы и эвристики	1	4	ОПК-2	Л1.1 Л1.2Л2.1 Л2.2 Л2.3Л3.1 Э1 Э3	4	работа в малых группах ОПК-2.1
1.11	Ср	Машины Тьюринга	1	12	УК-2 ОПК-2	Л1.1 Л1.2Л2.2 Л2.3Л3.1 Э1 Э2 Э3	0	УК-2.1 ОПК-2.1
1.12	Ср	Рекурсивные функции	1	13	УК-2 ОПК-2	Л1.1 Л1.2Л2.1 Л2.3Л3.1 Э1 Э2 Э3	0	УК-2.1 ОПК-2.2
1.13	Ср	Марковские подстановки	1	14	УК-2 ОПК-2	Л1.1 Л1.2Л2.1 Л2.2 Л2.3Л3.1 Э1 Э3	0	УК-2.1 ОПК-2.1
1.14	Экзамен	Теоретические основы теории алгоритмов	1	54	УК-2 ОПК-2	Л1.1 Л1.2Л2.1Л3.1 Э1 Э3	0	УК-2.1 ОПК-2.1,ОПК-2.2
	Раздел	<b>Раздел 2. Динамические структуры данных и алгоритмы их обработки</b>						
2.1	Лек	Линейные структуры данных	2	6	УК-2	Л1.2 Л1.3Л2.1 Э3	0	УК-2.1
2.2	Лек	Обзор динамических структур данных	2	2	УК-2	Л1.2 Л1.3Л2.1 Э2 Э3	0	УК-2.1
2.3	Лек	Роль структур данных при проектировании программ	2	6	УК-2	Л1.1 Л1.2 Л1.3Л2.1 Э3	4	проблемная лекция УК-2.1
2.4	Лек	Алгоритмы обработки массивов	2	4	УК-2	Л1.2 Л1.3 Э3	4	проблемная лекция УК-2.1
2.5	Лек	Иерархические структуры данных	2	6	УК-2 ОПК-2	Л1.2 Л1.3Л2.1 Э3	0	УК-2.1, ОПК-2.1, ОПК-2.2
2.6	Лек	Сетевые структуры данных	2	6	УК-2	Л1.2 Л1.3Л2.1 Э3	0	УК-2.1
2.7	Лек	Табличные структуры данных	2	6	УК-2	Л1.2 Л1.3 Э3	0	УК-2.1
2.8	Лаб	Модель предметной области	2	8	ОПК-2	Л1.2Л2.1 Э2 Э3	2	работа в малых группах ОПК-2.1

2.9	Лаб	Линейные структуры данных	2	14	ОПК-2	Л1.2Л2.1 Э3	4	работа в малых группах ОПК-2.1
2.10	Лаб	Нелинейные структуры данных	2	14	ОПК-2	Л1.2Л2.1 Э1 Э3	2	работа в малых группах ОПК-2.1
2.11	Ср	Модель предметной области	2	20	УК-2 ОПК-2	Л1.2 Э3	0	УК-2.1 ОПК-2.2
2.12	Ср	Линейные структуры данных	2	20	УК-2 ОПК-2	Л1.2Л2.1 Э1 Э3	0	УК-2.1 ОПК-2.1
2.13	Ср	Нелинейные структуры данных	2	16	УК-2 ОПК-2	Л1.2Л2.1 Э1	0	УК-2.1 ОПК-2.1
2.14	Ср	Алгоритмы обработки массивов	2	16	УК-2 ОПК-2	Л1.2Л3.1 Э3	0	УК-2.1 ОПК-2.1
2.15	КР	Динамические структуры данных и алгоритмы их обработки	2	18	УК-2 ОПК-2	Л1.1 Л1.2Л2.1 Э3	0	УК-2.1 ОПК-2.1
2.16	Экзамен	Динамические структуры данных и алгоритмы их обработки	2	18	УК-2 ОПК-2	Л1.1 Л1.2Л2.1Л3.1 Э3	0	УК-2.1 ОПК-2.1, ОПК-2.2

### 5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Образовательные технологии с использованием активных методов обучения (лекция – беседа, лекция – дискуссия, проблемная лекция, лекция-визуализация, лекция с заранее запланированными ошибками, лекция – пресс-конференция, лекция с разбором конкретных ситуаций, лекция-консультация, занятия с применением затрудняющих условий, методы группового решения творческих задач, метод развивающейся кооперации)

Традиционная (репродуктивная) технология (преподаватель знакомит обучающихся с порядком выполнения задания, наблюдает за выполнением и при необходимости корректирует работу обучающихся)

Технология коллективного взаимодействия (работа в малых группах) (самостоятельное изучение обучающимися нового материала посредством сотрудничества в малых группах, дает возможность всем участникам участвовать в работе, практиковать навыки сотрудничества, межличностного общения)

Образовательные технологии с использованием интерактивных методов обучения (круглый стол (дискуссия, дебаты), семинар - исследование, семинар «Пресс – антипресс», мозговой штурм (брейнсторм, мозговая атака), деловые, имитационные, операционные и ролевые игры, case-study (анализ конкретных ситуаций, ситуационный анализ), мастер класс, дидактические игры)

Технология компьютерного обучения (использование в учебном процессе компьютерных технологий и предоставляемых ими возможностей (электронные библиотеки, онлайн тесты, практические задания и т.д.))

### 6. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

#### 6.1. Контрольные вопросы и задания

Раздел 1. Теоретические основы теории алгоритмов

Задания для практических работ.

1. Разработайте инструкции на тему:

- как отучить кота портить мебель;
- как приготовить яичницу.

Запишите их в виде алгоритмов.

2. Составьте алгоритм деления отрезка на  $k$  равных частей.

3. Опишите алгоритм построения касательной к окружности, проведенной через заданную точку, лежащую вне этой окружности.

4. Предложите алгоритм поиска элемента:

- а) в неупорядоченном массиве;
- б) в полностью упорядоченном массиве без повторяющихся элементов.

5. Опишите алгоритм Евклида для нахождения наибольшего общего делителя.

6. Опишите алгоритм, определяющий для произвольного натурального числа  $N$ , является ли оно составным.

7. Перестановкой порядка  $n$  называется последовательность из попарно различных целых положительных чисел  $(p_1, p_2, \dots, p_n)$ , где  $1 < p_i < n$ . Будем говорить, что перестановка  $(q_1, q_2, \dots, q_n)$  лексикографически меньше перестановки  $(p_1, p_2, \dots, p_n)$ , если существует такое  $i$ , что  $q_i < p_i$ , а для любого  $j < i$  выполняется условие  $p_j = q_j$ . Циклическим сдвигом на  $k$  перестановки  $(p_1, p_2, \dots, p_n)$  называется последовательность  $(p_{k+1}, p_{k+2}, \dots, p_n, p_1, \dots, p_k)$ . Отметим, что любой циклический сдвиг перестановки также является перестановкой.

Найдите наименьший лексикографически циклический сдвиг заданной перестановки.

8. Определите, какие математические функции задаются следующими схемами:
- a)  $f(0,x)=O(x)$ ,  
 $f(y+1,x)=f(y,x)+x$ ;
- b)  $f(0,x)=\square(O(x))$ ,  
 $f(y+1,x)=f(y,x)\cdot x$
- c)  $f(0)=1$ ,  
 $f(y+1)=y\cdot f(y)+y$ .
9. Докажите, что любая примитивно рекурсивная функция всюду определена.
10. Докажите, что если функция  $f(x_1,x_2,\dots, x_n)$  примитивно рекурсивна, то и следующие функции примитивно рекурсивны:
- a)  $g(x_1,x_2,\dots, x_n)=f(x_2,x_1,\dots, x_n)$  (перестановка аргументов);
- b)  $p(x_1,x_2,\dots, x_n)=f(x_2,\dots, x_n,x_1)$  (циклическая перестановка аргументов);
- c)  $h(x_1,x_2,\dots, x_n, x_{n+1})=f(x_1,x_2,\dots, x_n)$  (введение фиктивного аргумента);
11. Какие функции получаются из простейших с помощью лишь суперпозиций?
12. Докажите, что из  $O(x)$  и  $\square$  с помощью суперпозиции и примитивной рекурсии нельзя получить функции  $x+1$  и  $2x$ .
13. Какая функция получается из  $\phi$  и  $\psi$  с помощью схемы примитивной рекурсии, если
- a)  $\phi(x)=x$ ,  $\psi(x,y,z) = zy$ ;
- б)  $\phi(x)=x$ ,  $\psi(x,y,z)= ux$ ;
- в)  $\phi(x)=x$ ,  $\psi(x,y,z) = xz$  ?
14. Докажите, что следующие функции примитивно рекурсивны:
- a) ;
- b) ;
- c) ;
- d) ;
- e)  $|x-y|$ ;
- f)  $\max(x,y)$ ;
- g) – частное от деления  $x$  на  $y$  (здесь );
- h)  $\text{rest}(x,y)$  – остаток от деления  $x$  на  $y$  (здесь  $\text{rest}(x,0)=0$ );
- i)  $\tau(x)$  – количество делителей числа  $x$ , где  $\tau(0)=0$ ;
- j)  $\sigma(x)$  – сумма делителей числа  $x$ , где  $\sigma(0)=0$ ;
- k)  $\text{IH}(x)$  – число простых делителей числа  $x$ ;
- l)  $\pi(x)$  – число простых чисел, не превосходящих  $x$ ;
- m)  $k(x,y) = \text{НОК}(x,y)$ , где  $k(x,0)=k(0,y)=0$ ;
- n) .
15. Докажите, что следующие функции частично рекурсивны:
- a)  $f(x,y)=$  ;
- b)  $f(x,y)=$  ;
- c)  $f(x,y)=$  .
16. Докажите, что если функции  $f$  и  $g$  частично рекурсивны, то следующие функции частично рекурсивны:
- a)  $\mu\{f(x_1,x_2,\dots, x_n, y)=g(x_1,x_2,\dots, x_n, y)\}$
- б)  $\mu\{f(x_1,x_2,\dots, x_n, y) \square g(x_1,x_2,\dots, x_n, y)\}$
- в)  $\mu\{f(x_1,x_2,\dots, x_n, y) \square g(x_1,x_2,\dots, x_n, y)\}$
- г)  $\mu\{f(x_1,x_2,\dots, x_n, y) < g(x_1,x_2,\dots, x_n, y)\}$
17. Пусть  $e=a_0.a_1a_2\dots$  - разложение числа  $e$  в бесконечную десятичную дробь. Докажите, что  $a_n$  – общерекурсивная функция от  $n$ .
18. Покажите, что следующие функции являются частично ( примитивно) рекурсивными:
- a)  $\text{rt}(n,x)=\lfloor \sqrt[n]{x} \rfloor$  - целая часть корня  $n$ -й степени из  $x$ ;
- b)  $\log(i, x) = \lfloor \log_i x \rfloor$ , где  $\log(i,x) = 0$  при  $i \in \{0,1\}$  или  $x = 0$ ;
- c) ;
- d)  $\text{pn}(k) = k$ -е простое число в порядке возрастания, например,  $\text{pn}(0)=0$ ,  $\text{pn}(1)=2$ ,  $\text{pn}(2)=3$ ,  $\text{pn}(3)=5$  .

#### Контрольные вопросы к практическим работам.

1. Что такое функциональная схема Тьюринга?
2. Что называется внешним алфавитом?
3. Расшифруйте строки Вашей программы.
4. Сформулируйте тезис Тьюринга.
5. Что называется Марковской подстановкой?
6. Что такое расширение алфавита и зачем оно нужно?
7. Как называется подстановка  $P_i \square Q_i$  ?
8. Как называется подстановка  $P_i \square \square Q_i$  ?
9. Что называется нормальным Марковским алгоритмом?
10. В каких случаях говорят, что алфавит неприменим к слову  $V$ ?
11. Опишите работу Вашего алгоритма.
12. К каким словам он не применим?
13. Какие функции называются нормально вычислимыми?

#### Раздел 2. Динамические структуры данных и алгоритмы их обработки.

Вопросы для защиты лабораторных работ.

1. Найти дубликаты в списке чисел с использованием дерева поиска.
2. На основе операции обхода сверху вниз реализовать операцию, определяющую, подобны ли два бинарных дерева (два бинарных дерева подобны, если они оба пусты, либо их левые и правые поддеревья подобны).
3. По бесскобочной постфиксной записи арифметического выражения с операндами, записанными в виде строк символов, построить дерево выражения и получить полноскобочное инфиксное выражение.
4. Реализовать операцию определения уровня узла с заданным указателем в бинарном дереве, построить дерево с узлами — символами и определить минимальный и максимальный уровни листа.
5. На основе процедуры обхода сверху вниз реализовать операцию поиска узла с заданным значением в дереве, не являющемся деревом поиска. Построить дерево минимальной высоты с элементами-символами. Используя операции Brother и Value, найти адрес брата узла с введенным с терминала значением и вывести его значение.
6. Построить дерево поиска с элементами — строками. Используя операции Addr, Father и Value, найти узел, являющийся самым «молодым» общим предком двух заданных узлов, и вывести его значение.
7. По постфиксной записи арифметического выражения с операндами-строками построить дерево выражения и получить инфиксную запись выражения, содержащую только необходимые скобки.
8. Реализовать дерево поиска, содержащее элементы с одинаковыми ключами. Операция Addr возвращает адрес первого элемента с заданным ключом. Операция Delete исключает первый элемент с заданным ключом.
9. Построить дерево поиска с элементами — вещественными числами. Определить количество элементов дерева на каждом уровне. Удалить элементы с заданными значениями.
10. Построить дерево поиска с элементами — числами. С использованием операций AcIcIг и Oe^eБей найти узел с заданным значением и исключить его левое поддерево. Вывести число узлов в дереве до и после исключения.
11. На основе просмотра текста построить дерево поиска, элементами которого являются символы, а ключами — количества вхождений этих символов в текст. Исключить из дерева символы с заданным числом повторений.
12. Построить дерево поиска с элементами — строками. Исключить из дерева все узлы с заданными словами. При каждом исключении выводить текущее состояние дерева и сообщение о том, каким потомком является исключаемый узел: левым (операция I8Бей) или правым (HЛ^M).
13. На основе процедуры обхода дерева слева направо реализовать операцию поиска узла с заданным значением в дереве, не являющемся деревом поиска. Из двух последовательностей символов построить два бинарных дерева минимальной высоты. В первом дереве найти элемент с заданным значением и подключить второе дерево в качестве его левого поддерева, если оно пусто, или левого поддерева первого из его крайних левых потомков, имеющих пустое левое поддерево.
14. На основе операции обхода сверху вниз реализовать операцию, определяющую, являются ли два бинарных дерева зеркально подобными (два бинарных дерева зеркально подобны, если они оба либо пусты, либо левое поддерево каждого дерева подобно правому дереву другого).
15. По бесскобочной постфиксной записи арифметического выражения с операндами — строками построить дерево выражения и получить полноскобочное инфиксное выражение.
16. По бесскобочной префиксной записи арифметического выражения с операндами — строками построить дерево выражения и получить выражение в инфиксной записи, содержащее только необходимые скобки.
17. Реализовать дерево поиска, содержащее элементы с одинаковыми ключами. Операция AbбгPцз! возвращает адрес первого элемента с заданным ключом. Операция OeIeIeAll исключает все элементы с заданным ключом.
18. Построить бинарное дерево с элементами — символами. Вывести элементы дерева по уровням.
19. Реализовать дерево выражения с операндами — числами и написать метод вычисления значения арифметического выражения. По бесскобочной префиксной записи построить дерево выражения и вычислить его значение.
20. Реализовать операцию определения уровня узла с заданным указателем в дереве поиска, построить дерево с заданными числовыми значениями узлов и определить минимальный и максимальный уровни листа.
21. На основе процедуры обхода дерева слева направо реализовать операцию поиска узла с заданным значением в дереве, не являющемся деревом поиска. Построить дерево минимальной высоты с числовыми значениями узлов. Используя необходимые операции, удалить из дерева поддерево, корнем которого является отец символа с заданным значением.
22. Построить два дерева поиска для студентов двух групп с полученными на экзаменах оценками. Сформировать дерево для студентов, получивших отличные оценки, причем расположить их в дереве по алфавиту.
23. Реализовать дерево поиска, содержащее элементы с одинаковыми ключами. Операция AddrLast возвращает адрес последнего элемента с заданным ключом. Операция DeleteFirst исключает первый элемент с заданным ключом.
24. С использованием дерева поиска удалить из заданного текста дубликаты слов.
25. Построить бинарное дерево с элементами — словами. Сформировать предложение из слов дерева, получающееся при обходе слева направо. Удалить поддерево, начинающееся с заданного слова.
26. Построить дерево поиска с элементами — целыми числами. Удалить из дерева элементы с заданным значением, а все отрицательные элементы заменить нулями.
27. Построить дерево поиска с заданными числовыми значениями. С использованием операций Addr и Father найти узел с заданным значением и удалить из дерева поиска отца этого узла.
28. Построить дерево поиска с элементами — символами. Определить число повторяющихся символов в дереве и удалить дубликаты.
29. Реализовать дерево поиска, содержащее элементы с одинаковыми ключами. Операция AddrLast возвращает адрес последнего элемента с заданным ключом. Операция DeleteAll исключает все элементы с заданным ключом.

## Раздел 2 Динамические структуры данных и алгоритмы их обработки

## Курсовая работа

Цель: обобщить и углубить знания по дисциплине алгоритмы и структуры данных

Структура: теоретическая информация, предназначенная для самостоятельного изучения

Практическая часть: индивидуальное задание на тему, предоставленную преподавателем.

Рекомендуемый объем 30-35 страниц

Темы для курсовых работ:

- 1 Генераторы псевдослучайных чисел.
- 2 Умножение больших чисел.
- 3 Поиск пути в графе.
- 4 Поиск подстроки в строке.
- 5 Поиск наибольшей общей подпоследовательности двух строк.
- 6 Алгоритмы умножения матриц.
- 7 Задача коммивояжера.
- 8 Поиск паросочетаний.
- 9 Решения систем линейных уравнений.
- 10 Факторизация чисел.
- 11 Обход дерева.
- 12 Поиск выпуклой оболочки.
- 13 Вычисление расстояния между строками.
- 14 Триангуляция многоугольника.
- 15 Нахождение простых чисел в заданном диапазоне.
- 16 Задача принадлежности точки выпуклому многоугольнику.
- 17 Поиск всех точек пересечения отрезков на плоскости.

Защита курсовой работы: открытая, перед учебной группой, обязательна презентация. Защита проводится только при наличии проверенного преподавателем текста курсовой работы.

### 6.3. Фонд оценочных средств

Вопросы к экзамену:

#### Раздел 1. Теоретические основы теории алгоритмов

1. Предмет изучения теории алгоритмов. Алгоритм и его свойства. Массовая задача.
2. Способы описания алгоритма. Примеры алгоритмов.
3. Регулярные языки. Грамматики.
4. Конечные автоматы.
5. Машина Тьюринга.
6. Понятие рекурсивной функции. Простейшие функции. Операции над функциями..
7. Марковские подстановки.
8. Проблема распознавания применимости и самоприменимости.
9. Виды эффективности алгоритма. Экспериментальная оценка сложности.
10. Теоремы о соотношении пространственной и временной эффективности.
11. Сложность задачи. Классы асимптотической сложности.
12. Примеры NP- сложных задач . Основная проблема теории сложности.
13. Полиномиальная сводимость. Задача оптимизации.
14. Приближенные алгоритмы. Мультипликативная ошибка. Эвристики.

#### Раздел 2. Динамические структуры данных и алгоритмы их обработки

1. Уровни представления данных
2. Концепция типа данных. Назначение типов. Элементарные данные, их представление в памяти.
3. Представление чисел с плавающей точкой
4. Структуры (записи) . Назначение, представление в памяти компьютера.
5. Основные этапы разработки алгоритмов и программ
6. Оценки эффективности (сложности) алгоритма: пространственная, временная, коммуникативная. Теорема о соотношении пространственной и временной сложности
7. Асимптотические оценки сложности. Основные определения: асимптотическая оценка, оценки в худшем, лучшем, среднем.
8. Классы сложности алгоритмов: обозначения, примеры
9. Сложность задачи и сложность алгоритма. Классы P- и NP- сложных задач. Примеры. Основная проблема сложности.
10. Приближенные алгоритмы и эвристики: задача оптимизации, целевая функция, точный и приближенный алгоритм. Мультипликативная ошибка алгоритма. Пример приближенного алгоритма: задача о рюкзаке
11. Приближенный алгоритм задачи о камнях (задачи о машинах): онлайн-алгоритм, его мультипликативная ошибка. С доказательством
12. Жадные алгоритмы. Жадный алгоритм задачи о покрытии множеств (о выборе переводчиков)
13. Вероятностные алгоритмы. Алгоритм анализа числа на простоту, основанный на теореме Ферма. Односторонняя ошибка.
14. Алгоритмы на графах и их сложность: задача коммивояжера, задача об остове минимального веса, поиск кратчайшего пути в графе



15.	Стеки: определение, функционал, назначение. Реализация стека
16.	Парсинг арифметических выражений: алгоритм Дейкстры
17.	Очереди: виды очередей, назначение. Допустимые операции. Реализация очереди
18.	Списки: виды списков, назначение. Допустимые операции. Реализация очереди
19.	Графы: основные определения, виды графов. Матрицы смежности и инцидентности
20.	Графы: построение матрицы достижимости в графе
21.	Графы: представление графа списком. Преимущества и недостатки такого представления. Задачи. эффективно решаемые на таком представлении
22.	Деревья: основные понятия и определения. Виды деревьев. Задачи, решаемые с применением этой структуры данных
23.	Бинарные деревья: преобразование упорядоченного списка в дерево
24.	Обход бинарного дерева
25.	Сортирующее дерево. Алгоритмы сортировок, основанные на использовании деревьев.
26.	Коды Хаффмана
27.	Словари и хеш-таблицы. Хеш-функции. Открытое и закрытое хеширование

#### 6.4. Перечень видов оценочных средств

Отчет по лабораторным работам.
Отчет по практическим работам.
Задания для практических работ.
Вопросы для защиты лабораторных работ.
Индивидуальное задание на курсовую работу.
Вопросы к экзамену.

### 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

#### 7.1. Рекомендуемая литература

##### 7.1.1. Основная литература

	Авторы,	Заглавие	Издательство,	Кол-во	Эл. адрес
Л1. 1	Макконелл Дж.	Основы современных алгоритмов: Учеб. пособие для вузов	Москва: Техносфера, 2006	27	
Л1. 2	Симонович С.В.	Информатика. Базовый курс: учебник для бакалавров и специалистов	Санкт-Петербург: Питер, 2014	76	
Л1. 3	Царёв Р. Ю., Прокопенко А. В.	Алгоритмы и структуры данных (CDIO): учебник	Красноярск: СФУ, 2016	1	<a href="http://biblioclub.ru/index.php?page=book&amp;id=497016">http://biblioclub.ru/index.php?page=book&amp;id=497016</a>

##### 7.1.2. Дополнительная литература

	Авторы,	Заглавие	Издательство,	Кол-во	Эл. адрес
Л2. 1	Серебряная Л.В., Марина И.М.	Структуры и алгоритмы обработки данных: учебно-методическое пособие	Минск: БГУИР, 2013	1	<a href="http://ecat.brstu.ru/catalog/Ресурсы%20свободного%20доступа/Серебряная%20Л.В.%20Структуры%20и%20алгоритмы%20обработки%20данных.%20Учеб.-метод.%20пособие.%202013.pdf">http://ecat.brstu.ru/catalog/Ресурсы%20свободного%20доступа/Серебряная%20Л.В.%20Структуры%20и%20алгоритмы%20обработки%20данных.%20Учеб.-метод.%20пособие.%202013.pdf</a>
Л2. 2	Комлева Н.В.	Структуры и алгоритмы компьютерной обработки данных: Учебное пособие, руководство по дисциплине, практикум, тесты, учебная программа	Москва: Московский государственный университет экономики, статистики и информатики, 2004	1	<a href="http://biblioclub.ru/index.php?page=book_red&amp;id=93226">http://biblioclub.ru/index.php?page=book_red&amp;id=93226</a>
Л2. 3	Абрамов С.А.	Лекции о сложности алгоритмов: учебное пособие	Москва : МЦНМО, 2009	1	<a href="http://biblioclub.ru/index.php?page=book&amp;id=63276">http://biblioclub.ru/index.php?page=book&amp;id=63276</a>

##### 7.1.3. Методические разработки

	Авторы,	Заглавие	Издательство,	Кол-во	Эл. адрес
Л3. 1	Ратинская Е.В.	Теория алгоритмов: Учебное пособие	Братск: БрГУ, 2011	68	

#### 7.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	«Университетская библиотека online»	<a href="http://biblioclub.ru">http://biblioclub.ru</a>
Э2	Электронный каталог библиотеки БрГУ	<a href="https://irbis.brstu.ru/CGI/irbis64r_15/cgiirbis_64.exe?LNG=&amp;C21COM=F&amp;I21DBN=BOOK&amp;P21DBN=BOOK&amp;S21CNR=&amp;Z21ID=">https://irbis.brstu.ru/CGI/irbis64r_15/cgiirbis_64.exe?LNG=&amp;C21COM=F&amp;I21DBN=BOOK&amp;P21DBN=BOOK&amp;S21CNR=&amp;Z21ID=</a>

ЭЗ	Электронная библиотека БрГУ	<a href="https://ecat.brstu.ru/catalog">https://ecat.brstu.ru/catalog</a>
<b>7.3.1 Перечень программного обеспечения</b>		
7.3.1.1	GNU gcc	
7.3.1.2	ОС Linux	
7.3.1.3	LibreOffice	
<b>7.3.2 Перечень информационных справочных систем</b>		
7.3.2.1	Электронная библиотека БрГУ	
7.3.2.2	Электронный каталог библиотеки БрГУ	
7.3.2.3	«Университетская библиотека online»	
<b>8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)</b>		
A1203	Лаборатория параллельных вычислений	Персональный компьютер i5-2500/H67/4Gb/500Gb(Монитор TFT19 Samsung E1920NR), интерактивная доска SMART Board X885ix со встроенным проектором UX 60,доска магнитно-маркерная .
0002*	лекционная аудитория	Учебная мебель
0002*	лекционная аудитория	Учебная мебель
1001	читальный зал №3	Учебная мебель, Оборудование 15- CPU 5000/RAM 2Gb/HDD (Монитор TFT 19 LG 1953S-SF);принтер HP LaserJet P3005
0001*	аудитория для практических занятий	Учебная мебель
1001	читальный зал №3	Учебная мебель, Оборудование 15- CPU 5000/RAM 2Gb/HDD (Монитор TFT 19 LG 1953S-SF);принтер HP LaserJet P3005
<b>9. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)</b>		
<p>Обучающийся должен разработать собственный режим равномерного освоения дисциплины.</p> <p>Подготовка студента к предстоящей лекции включает в себя ряд важных познавательных-практических этапов: чтение записей, сделанных в процессе слушания и конспектирования предыдущей лекции, вынесение на поля всего, что требуется при дальнейшей работе с конспектом и учебником;техническое оформление записей (подчеркивание, выделение главного, выводов, доказательств);</p> <p>выполнение практических заданий преподавателя;знакомство с материалом предстоящей лекции по учебнику и дополнительной литературе.</p> <p>Успешность выполнения лабораторных работ и практических заданий определяется подготовкой к ним.</p> <p>Подготовка к лабораторным работам и практическим занятиям содержит:</p> <ul style="list-style-type: none"> <li>- изучение теоретического материала, содержащегося в учебной литературе, изучение лекционного материала,</li> <li>- знакомство с заданиями;</li> <li>- составление плана выполнения;</li> <li>- реализация;</li> <li>- написание отчета.</li> </ul> <p>В процессе изучения дисциплины студент должен выполнить курсовую работу, основной целью которых является проверка его знаний, умений и навыков, полученных при изучении дисциплины.</p> <p>Наиболее продуктивной является самостоятельная работа. Она складывается из чтения учебников и методических пособий, решения задач, выполнения контрольных заданий.</p> <p>Студент должен помнить, что только при систематической и упорной самостоятельной работе можно качественно освоить учебный материал.</p> <p>Завершающим этапом изучения данной дисциплины в соответствии с учебным планом является сдача экзамена. На экзамене студент должен: проявить умение применять теоретические сведения к решению задач построение и анализ алгоритмов; знание теоретических основ курса на уровне определений, теорем, формул; умение выбирать методы анализа и оценки выбранных решений.</p>		