

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Сыров Игорь Анатольевич  
Должность: Директор  
Дата подписания: 24.06.2022 14:07:36  
Уникальный программный ключ:  
b683afe664d7e9f64175886cf9626a196149ad56

СТЕРЛИТАМАКСКИЙ ФИЛИАЛ  
ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО  
УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ  
«БАШКИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет Математики и информационных технологий  
Кафедра Прикладной информатики и программирования

**Аннотация рабочей программы дисциплины (модуля)**

дисциплина ***Б1.В.05 Теоретические основы информатики***  
часть, формируемая участниками образовательных отношений

Направление  
**44.03.05 Педагогическое образование (с двумя профилями подготовки)**  
код наименование направления

Программа  
**Математика, Информатика**

Форма обучения  
**Очная**  
Для поступивших на обучение в  
**2019 г.**

**1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций**

| <b>Формируемая компетенция (с указанием кода)</b>  | <b>Код и наименование индикатора достижения компетенции</b>  | <b>Результаты обучения по дисциплине (модулю)</b>  |
|--|--|--|
| ПК-1. Способен разрабатывать образовательные программы по предметам в соответствии с требованиями образовательных стандартов | ПК-1.1. Знать: теорию и методику преподавания предмета; программы и учебники по преподаваемому предмету; основные нормативно-правовые документы общего образования.  | Обучающийся должен знать: основные кодировки текстовой информации в компьютере; правила перевода чисел в различные системы счисления.  |
|  | ПК-1.2. Уметь: анализировать педагогические ситуации с точки зрения требований к содержанию и качеству образования (требования ФГОС); анализировать результаты проектирования урока (планы, планы-конспекты, сценарии, методические разработки уроков, далее – проекты) в контексте требований к содержанию и качеству образования (требования ФГОС); разрабатывать и реализовывать индивидуальные программы развития и индивидуально-ориентированные образовательные программы с учетом личностных и возрастных особенностей обучающихся. | Обучающийся должен уметь: определять количество информации; использовать знания по теории информации, теории кодирования, арифметические и логические основы ЭВМ в профессиональной деятельности; осуществлять перевод числовой информации из одной системы счисления в другую.                            |
|  | ПК-1.3. Владеть: логико-дидактическим анализом (ЛДА) содержания школьных учебников, навыками осуществления профессиональной деятельности в соответствии с требованиями ФГОС основного общего, среднего общего образования, разработки и реализация программ учебных дисциплин в рамках основной общеобразовательной программы.   | Обучающийся должен владеть: основными методами, способами и средствами получения, хранения, переработки информации; основными способами преобразования чисел в различных системах счисления и их представления в памяти ЭВМ; навыками выполнения арифметических действий в позиционных системах счисления. |

## 2. Цели и место дисциплины (модуля) в структуре образовательной программы

Цели изучения дисциплины:

Дисциплина Теоретические основы информатики относится к части, формируемой участниками образовательных отношений.

Цели изучения дисциплины:

1. Развить способность использовать естественнонаучные и математические знания для ориентирования в современном информационном пространстве;
2. Мотивировать готовность реализовывать образовательные программы по учебным предметам в соответствии с требованиями образовательных стандартов.

Дисциплина изучается на 2, 3 курсах в 4, 5 семестрах

## 3. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических или астрономических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость (объем) дисциплины составляет 8 зач. ед., 288 акад. ч.

| Объем дисциплины   | Всего часов          |
|--|----------------------|
|  | Очная форма обучения |
| Общая трудоемкость дисциплины                            | 288                  |
| Учебных часов на контактную работу с преподавателем:     |                      |
| лекций   | 28                   |
| практических (семинарских)                               | 52                   |
| лабораторных   | 32                   |
| другие формы контактной работы (ФКР)                     | 1,4                  |
| Учебных часов на контроль (включая часы подготовки):     | 34,8                 |
| зачет  |                      |
| экзамен  |                      |
| Учебных часов на самостоятельную работу обучающихся (СР) | 139,8                |

| Формы контроля | Семестры |
|----------------|----------|
| зачет          | 4        |
| экзамен        | 5        |

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

### 4.1. Разделы дисциплины и трудоемкость по видам учебных занятий (в академических часах)

| № п/п    | Наименование раздела / темы дисциплины | Виды учебных занятий, включая самостоятельную работу обучающихся и трудоемкость (в часах) |           |           |           |
|----------|--|---|-----------|-----------|-----------|
|          |  | Контактная работа с преподавателем  |           |           | СР        |
|          |  | Лек   | Пр/Сем    | Лаб       |           |
| 1.1      | Теория информации.                     | 7   | 10        | 8         | 29,8      |
| 1.2      | Теория кодирования.                    | 7   | 10        | 8         | 30        |
| <b>2</b> | <b>5 семестр</b>                       | <b>14</b>   | <b>32</b> | <b>16</b> | <b>80</b> |
| 2.1      | Теория алгоритмов.                     | 7   | 16        | 8         | 40        |
| 2.2      | Основные не вычислимые                 | 7   | 16        | 8         | 40        |

|          |                  |           |           |           |              |
|----------|------------------|-----------|-----------|-----------|--------------|
|          | алгоритмы.       |           |           |           |              |
| <b>1</b> | <b>4 семестр</b> | <b>14</b> | <b>20</b> | <b>16</b> | <b>59,8</b>  |
|          | <b>Итого</b>     | <b>28</b> | <b>52</b> | <b>32</b> | <b>139,8</b> |

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

Курс лабораторных занятий

| №        | Наименование раздела / темы дисциплины | Содержание   |
|----------|--|--|
| 1.1      | Теория информации.                     | <p>1. Имеется два текста на разных языках. Первый текст использует 32-символьный алфавит и содержит 200 символов, второй — 16-символьный алфавит и содержит 250 символов. Какой из текстов содержит большее количество информации и на сколько битов?</p> <p>2. Сколько символов в тексте, если мощность алфавита -- 64 символа, а объем информации, содержащейся в нем, — 1,5 килобайта?</p> <p>3. Шарик находится в одном из 64 ящичков. Сколько единиц информации будет содержать сообщение о том, где находится шарик?</p> <p>4. Рассчитать энтропию и избыточность алфавитного источника информации из табл.1 с объемом <math>N=77</math> (номер алфавита соответствует порядковому номеру студента в списке группы).</p>   |
| 1.2      | Теория кодирования.                    | <p>1. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, решили использовать неравномерный двоичный код, позволяющий однозначно декодировать двоичную последовательность, появляющуюся на приёмной стороне канала связи. Использовали код: А–1, Б–000, В–001, Г–011. Укажите, каким кодовым словом должна быть закодирована буква Д. Длина этого кодового слова должна быть наименьшей из всех возможных. Код должен удовлетворять свойству однозначного декодирования.</p> <p>2. Даны символы a, b, c, d, e, g с частотами <math>f(a) = 0,25</math>, <math>f(b) = 0,05</math>, <math>f(c) = 0,25</math>, <math>f(d) = 0,1</math>, <math>f(e) = 0,25</math>, <math>f(g) = 0,1</math>. Построить эффективный код методом Хаффмана и определите, какой код получил символ d.</p> <p>3. Выполнить проверку на оптимальность двоичного кода, полученного методом Шеннона - Фано для алфавита <math>A = \{a_1, a_2, a_3, a_4, a_5, a_6\}</math> с вероятностями появления отдельных символов (0,25 - 0,3 - 0,15 - 0,2 - 0,05 - 0,05) соответственно.</p> <p>4. Блочный помехоустойчивый код имеет длину блока 8 бит. Его избыточность равна 25%. Чему равно число информационных разрядов в нем?</p> <p>5. Вычислите длину кодового слова при неравномерном кодировании, если алфавит состоит из символов <math>A = \{a, b, c, d, e\}</math>, вероятность <math>p(a)=0,5</math>, <math>p(b)=0,2</math>, <math>p(c)=0,1</math>, <math>p(d)=0,15</math>, <math>p(e)=0,05</math><br/> В: a=0, b=10, c=1110, d=110, e=1111</p> |
| <b>2</b> | <b>5 семестр</b>                       |  |
| 2.1      | Теория алгоритмов.                     | <p>1. Дан массив меток. Каретка располагается где-то над массивом, но не над крайними метками. Стереть все метки, кроме крайних, и поставить каретку в исходное положение.</p>   |

|  |  |
|--|--|
|  | <p>2. На ленте машины Поста расположен массив из <math>n</math> меток (метки расположены через пробел). Нужно сжать массив так, чтобы все <math>n</math> меток занимали <math>n</math> расположенных подряд ячеек.</p> <p>3. Дано несколько массивов меток. Удалить четные массивы. Каретка находится над первым массивом.</p> <p>4. На ленте машины Поста расположено <math>n</math> массивов меток, отделенных друг от друга свободной ячейкой. Каретка находится над крайней левой меткой первого массива. Определить количество массивов.</p> <p>5. На ленте машины Поста расположен массив из <math>2n - 1</math> меток. Составить программу удаления средней метки массива.</p> <p>6. На ленте машины Поста расположен массив из <math>2n</math> ячеек. Составить программу, по которой машина Поста раздвинет на расстояние в одну ячейку две половины данного массива.</p> <p>7. На ленте расположены два массива разной длины. Каретка обозревает крайний элемент одного из них. Составьте программу для машины Поста, сравнивающую длины массивов и стирающую больший из них. Отдельно продумайте случай, когда длины массивов равны.</p> <p>8. На ленте машины Поста находятся два массива в <math>m</math> и <math>n</math> меток. Составить программу выяснения, одинаковы ли массивы по длине.</p> <p>9. Дано <math>N</math> массивов меток. Массивы разделены тремя пустыми ячейками. Количество меток в массиве не меньше двух. Если количество меток в массиве кратно трем, то стереть метки в этом массиве через одну, в противном случае стереть весь массив. Каретка находится над крайней левой меткой первого массива.</p> <p>10. На ленте машины Тьюринга содержится последовательность символов “+”. Напишите программу для машины Тьюринга, которая каждый второй символ “+” заменит на “-”. Замена начинается с правого конца последовательности. Автомат в состоянии <math>q_1</math> обозревает один из символов указанной последовательности. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>11. Дано число <math>n</math> в восьмеричной системе счисления. Разработать машину Тьюринга, которая увеличивала бы заданное число <math>n</math> на 1. Автомат в состоянии <math>q_1</math> обозревает некую цифру входного слова. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>12. Дана десятичная запись натурального числа <math>n &gt; 1</math>. Разработать машину Тьюринга, которая уменьшала бы заданное число <math>n</math> на 1. Автомат в состоянии <math>q_1</math> обозревает правую цифру числа. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>13. Дано натуральное число <math>n &gt; 1</math>. Разработать машину Тьюринга, которая уменьшала бы заданное число <math>n</math> на 1, при этом в выходном слове старшая цифра не должна быть 0. Например, если входным словом было “100”, то выходным словом должно быть “99”, а не “099”. Автомат в состоянии <math>q_1</math> обозревает правую цифру числа. Кроме самой программы-</p> |
|--|--|

|     |                                  |   |
|-----|----------------------------------|---|
|     |                                  | <p>таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>14. Дан массив из открывающих и закрывающих скобок. Построить машину Тьюринга, которая удаляла бы пары взаимных скобок, т.е. расположенных подряд “( )”. Например, дано “( ( ( ( ( ) ) ) ) )”, надо получить “) . . . ( ( ”. Автомат в состоянии q1 обозревает крайний левый символ строки. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>15. Дана строка из букв “a” и “b”. Разработать машину Тьюринга, которая переместит все буквы “a” в левую, а буквы “b” — в правую части строки. Автомат в состоянии q1 обозревает крайний левый символ строки. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>16. На ленте машины Тьюринга находится число, записанное в десятичной системе счисления. Умножить это число на 2. Автомат в состоянии q1 обозревает крайнюю левую цифру числа. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>17. Даны два натуральных числа m и n, представленные в унарной системе счисления. Соответствующие наборы символов “ ” разделены пустой клеткой. Автомат в состоянии q1 обозревает самый правый символ входной последовательности. Разработать машину Тьюринга, которая на ленте оставит сумму чисел m и n. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>18. Даны два натуральных числа m и n, представленных в унарной системе счисления. Соответствующие наборы символов “ ” разделены пустой клеткой. Автомат в состоянии q1 обозревает самый правый символ входной последовательности. Разработать машину Тьюринга, которая на ленте оставит разность чисел m и n. Известно, что <math>m &gt; n</math>. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> <p>19. На ленте машины Тьюринга находится десятичное число. Определить, делится ли это число на 5 без остатка. Если делится, то записать справа от числа слово “да”, иначе — “нет”. Автомат обозревает некую цифру входного числа. Кроме самой программы-таблицы, описать словами, что выполняется машиной в каждом состоянии.</p> |
| 2.2 | Основные невычислимые алгоритмы. | <p>1. Написать программу поиска элемента x в массиве из n элементов. Значение элемента x вводится с клавиатуры.</p> <p>2. Дано целое число x и массив <math>a[1..n]</math>, отсортированных в порядке не убывания чисел, то есть для любого <math>i</math>, что <math>a[i] \leq a[i+1]</math>. Найти такое <math>i</math>, что <math>a[i] \leq x &lt; a[i+1]</math> или сообщить, что элемента x в массиве нет.</p> <p>3. Заданы две строки s и x. Длина первой строки n, длина второй — m, причем <math>n \leq m</math>. Требуется ответить на вопрос, является ли строка x подстрокой строки s, при этом поиск подстроки должен обнаруживать первое вхождение x в s.</p> <p>4. Даны строка и подстрока. Составить программу, которая</p>  |

|  |   |
|--|---|
|  | <p>определяет вхождение подстроки в строку по алгоритму Бойера—Мура.</p> <p>5. Ниже представлена программа поиска подстроки в строке по алгоритму Кнута, Мориса, Пратта. Изучите идею метода и работу алгоритма.</p> <p>6. Поиск в упорядоченном по возрастанию массиве первого вхождения числа X</p> <p>7. Имеется массив <math>a[1..n]</math>, требуется найти элемент массива, равный P.</p> <p>8. Составить программу для поиска максимального элемента массива.</p> <p>9. Создать программу, выполняющую бинарный поиск заданного элемента в отсортированном массиве целых чисел, элементы которого имеют значения:<br/>20, 20, 19, 19, 19, 18, 17, 17, 12, 12, 11, 10, 9, 9, 5, 5, 3, 3, 2, 1.</p> <p>10. Создать программу, формирующую двумерный массив случайных чисел и вычисляющую значение среднего арифметического его элементов, больших, чем 20.</p> <p>11. Написать программу линейной сортировки массива M по убыванию.</p> <p>12. Найти максимальный элемент из <math>a[1], \dots, a[i]</math>. Сортировкой метода простого выбора.</p> <p>13. Отсортировать по возрастанию методом простого обмена массив из пяти элементов: 5 1 8 4 9.</p> <p>14. Отсортировать массив из 10 элементов по возрастанию методом прямого включения: 13 6 8 11 3 1 5 9 15 7.</p> <p>15. Отсортировать массив методом слияний. Первая часть массива (часть A) состоит из пяти элементов: ...358 1116..., а вторая часть (часть B) — из восьми элементов: ... 1 67 9 12 13 18 20... Оба массива-части упорядочены.</p> <p>16. Отсортировать массив методом «быстрой сортировки» состоящий из восьми элементов: 8 12 3 7 19 11 4 16.</p> <p>17. Отсортировать массив методом «пузырька» по убыванию.</p> <p>18. Отсортировать массив быстрой сортировкой по возрастанию дроблением массива на части.</p> <p>19. Рассортировать по убыванию все не четные числа одномерного массива с помощью метода "пузырька".</p> <p>20. Методом простого включения упорядочить (отсортировать) в порядке возрастания массив из 8 целых чисел (44, 55, 12, 42, 94, 18, 06, 67).</p> <p>21. Вычислить <math>(a! + b!)/a!</math>, используя рекурсивную функцию вычисления факториала.</p> <p>22. Вычислить <math>m!/(m + n)!</math>, используя рекурсивную функцию вычисления факториала.</p> <p>23. Вычислить <math>(1+2+3+4+5)/(1+2+3+4+5+6+7+8)</math>, используя рекурсивную функцию вычисления суммы первых n натуральных чисел.</p> <p>24. Составить рекурсивную функцию для вычисления <math>S = 2 + 4 + 6 + \dots + 2*n</math>.</p> <p>25. Составить рекурсивную функцию для вычисления <math>S = 5 + 10 + 15 + \dots + 5*n</math>.</p> <p>26. Составить рекурсивную функцию для вычисления</p> |
|--|---|

|          |                  |  |
|----------|------------------|--|
|          |                  | $P = 2 * 4 * 6 * \dots * 2 * n.$<br>27. Составить рекурсивную функцию вычисления n-го члена арифметической прогрессии 3, 7, ... и вывести первые 10 членов прогрессии.<br>28. Составить рекурсивную функцию вычисления n-го члена геометрической 1, 2, ... и вывести первые 8 членов прогрессии. |
| <b>1</b> | <b>4 семестр</b> |  |

Курс практических/семинарских занятий

| <b>№</b> | <b>Наименование раздела / темы дисциплины</b> | <b>Содержание</b>  |
|----------|---|--|
| 1.1      | Теория информации.                            | 1. Имеется два текста на разных языках. Первый текст использует 32-символьный алфавит и содержит 200 символов, второй — 16-символьный алфавит и содержит 250 символов. Какой из текстов содержит большее количество информации и на сколько битов?<br>2. Сколько символов в тексте, если мощность алфавита -- 64 символа, а объем информации, содержащейся в нем, — 1,5 килобайта?<br>3. Шарик находится в одном из 64 ящичков. Сколько единиц информации будет содержать сообщение о том, где находится шарик?<br>4. Рассчитать энтропию и избыточность алфавитного источника информации из табл.1 с объемом $N=77$ (номер алфавита соответствует порядковому номеру студента в списке группы).<br>5. Найти приближенное значение энтропии русского алфавита и его избыточность, воспользовавшись данными из табл.2, округленными до второго знака после запятой.   |
| 1.2      | Теория кодирования.                           | 1. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, используется неравномерный двоичный код, позволяющий однозначно декодировать полученную двоичную последовательность. Вот этот код: А – 0; Б – 100; В – 1010; Г – 111; Д – 110. Требуется сократить для одной из букв длину кодового слова так, чтобы код по-прежнему можно было декодировать однозначно. Коды остальных букв меняться не должны.<br>2. Объем информационного сообщения 12582912 битов выразить в килобайтах и мегабайтах.<br>3. Для кодирования некоторой последовательности, состоящей из букв А, Б, В, Г и Д, решили использовать неравномерный двоичный код, позволяющий однозначно декодировать двоичную последовательность, появляющуюся на приёмной стороне канала связи. Использовали код: А–1, Б–000, В–001, Г–011. Укажите, каким кодовым словом должна быть закодирована буква Д. Длина этого кодового слова должна быть наименьшей из всех возможных. Код должен удовлетворять свойству однозначного декодирования.<br>4. Даны символы a, b, c, d, e, g с частотами $f(a) = 0,25$ , $f(b) = 0,05$ , $f(c) = 0,25$ , $f(d) = 0,1$ , $f(e) = 0,25$ , $f(g) = 0,1$ . Построить эффективный код методом Шеннона-Фано и определите, какой |

|          |                    |   |
|----------|--------------------|---|
|          |                    | <p>код получил символ b.</p> <p>5. Выполнить проверку на оптимальность двоичного кода, полученного методом Шеннона - Фано для алфавита <math>A = \{a_1, a_2, a_3, a_4, a_5, a_6\}</math> с вероятностями появления отдельных символов (0,25 - 0,3 - 0,15 - 0,2 - 0,05 - 0,05) соответственно.</p> <p>6. Даны символы a, b, c, d, e, g с частотами <math>f(a) = 0,25, f(b) = 0,05, f(c) = 0,25, f(d) = 0,1, f(e) = 0,25, f(g) = 0,1</math>. Построить эффективный код методом Хаффмана и определите, какой код получил символ d.</p> <p>7. Блочный помехоустойчивый код имеет длину блока 8 бит. Его избыточность равна 25%. Чему равно число информационных разрядов в нем?</p> <p>8. Вычислите длину кодового слова при неравномерном кодировании, если алфавит состоит из символов <math>A = \{a, b, c, d, e\}</math>, вероятность <math>p(a)=0,5, p(b)=0,2, p(c)=0,1, p(d)=0,15, p(e)=0,05</math><br/> <math>V: a=0, b=10, c=1110, d=110, e=1111</math></p> <p>9. Пусть дан алфавит <math>A = \{A, B, C\}</math> и алфавит <math>V = \{0, 1\}</math>. <math>A=0, B=10, C=110</math>. Закодированное сообщение имеет вид 01010110. Раскодируйте сообщение.</p>  |
| <b>2</b> | <b>5 семестр</b>   |   |
| 2.1      | Теория алгоритмов. | <p>1. На ленте проставлена метка в одной-единственной ячейке. Каретка стоит на некотором расстоянии левее этой ячейки. Необходимо подвести каретку к ячейке, стереть метку и остановить каретку слева от этой ячейки.</p> <p>2. На ленте задан массив меток. Увеличить длину массива на 2 метки. Каретка находится либо слева от массива, либо над одной из ячеек самого массива.</p> <p>3. Даны два массива меток, которые находятся на некотором расстоянии друг от друга. Требуется соединить их в один массив. Каретка находится над крайней левой меткой первого массива.</p> <p>4. (анализ символов). <math>A = \{a, b, c\}</math>. Перенести первый символ непустого слова P в его конец.</p> <p>5. (перемещение автомата, замена символов). <math>A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}</math>. Пусть P – непустое слово; значит, P – это последовательность из десятичных цифр, т.е. запись неотрицательного целого числа в десятичной системе. Требуется получить на ленте запись числа, которое на 1 больше числа P.</p> <p>6. (сравнение символов, стирание слова). <math>A = \{a, b, c\}</math>. Если первый и последний символы (непустого) слова P одинаковы, тогда это слово не менять, а иначе заменить его на пустое слово.</p> <p>7. (вставка и удаление символов) <math>A = \{a, b, c, d\}</math>. В слове P требуется заменить первое вхождение под слова bb на ddd и удалить все вхождения символа c.</p> <p>8. (перестановка символов) <math>A = \{a, b\}</math>. Преобразовать слово P так, чтобы в его начале оказались все символы a, а в конце - все символы b.</p> <p>9. (фиксация спецзнаком заменяемого символа) <math>A = \{0, 1, 2, 3\}</math>. Пусть P - непустое слово. Трактую его как запись неотрицательного целого числа в четверичной системе</p> |

|     |                                    |   |
|-----|------------------------------------|---|
|     |                                    | счисления, требуется получить запись этого же числа, но в двоичной системе.<br>10. (использование спецзнака) $A=\{a,b\}$ . Удалить из непустого слова $P$ его первый символ. Пустое слово не менять.  |
| 2.2 | Основные не вычисляемые алгоритмы. | 1. Имеется массив $a[1..n]$ , требуется найти элемент массива, равный $P$ .<br>2. Составить программу для поиска максимального элемента массива.<br>3. Создать программу, формирующую двумерный массив случайных чисел и вычисляющую значение среднего арифметического его элементов, больших, чем 20.<br>4. Дана непустая последовательность натуральных чисел, за которой следует 0. Составить программу поиска в данной непустой последовательности порядкового номера наименьшего элемента.<br>5. Отсортировать массив методом «пузырька» по убыванию.<br>6. Отсортировать массив быстрой сортировкой по возрастанию дроблением массива на части.<br>7. Рассортировать по убыванию все не четные числа одномерного массива с помощью метода "пузырька".<br>8. Вычислить значения функции $f(x)=2\cos x+3$ , при $x=\{1; 4; 7,5; 20\}$ . Вывести результаты в два столбца: в первом – значения $x$ , во втором – значения $f(x)$ . Вычисления провести двумя способами: с помощью функции и процедуры.<br>9. Создать рекурсивную функцию поиска $i$ -го члена последовательности, заданной рекуррентной формулой $A_1=\text{const}1, A_2=\text{const}2, A_i=3A_{i-2}-A_{i-1}$ . Вывести через пробел значения рекурсивной функции при значениях аргумента от 1 до 10 включительно.<br>10. Вычислить сложность машины Тьюринга для вычисления функции $f(n)=2n+3$ , а также произвести быструю приближительную оценку сложности этой машины.<br>11. Найти пространственную и временную сложность алгоритма машины Тьюринга, вычисляющей в унарном коде функцию $f(x)=x^2$ |
| 1   | 4 семестр                          |   |

Курс лекционных занятий

| №   | Наименование раздела / темы дисциплины | Содержание   |
|-----|--|--|
| 1.1 | Теория информации.                     | Понятие информации. Сообщение и информация. Свойства информации. Действия с информацией. Измерение информации. Постулат Шеннона. Равновероятные исходы. Не равновероятные исходы.<br>Энтропия и алфавит. Свойства энтропии. Избыточность алфавита. Помехоустойчивость природных языков. Количество информации в сообщении. |
| 1.2 | Теория кодирования.                    | Кодирование информации. Связь и кодирование. Кодирование. Равномерные и неравномерные коды. Условие Фано. Количество и объем информации. Избыточность кода.  |

|          |                                   |   |
|----------|-----------------------------------|---|
|          |                                   | Оптимальное кодирование. Теорема кодирования. Метод Шеннона - Фано. Метод Хаффмана. Проверка кода на оптимальность. Блочное кодирование.  |
| <b>2</b> | <b>5 семестр</b>                  |   |
| 2.1      | Теория алгоритмов.                | Постановка проблемы. Понятие о машине Поста. Примеры построения машин Поста.<br>Понятие о машине Тьюринга. Тезис Тьюринга. Примеры построения машин Тьюринга.   |
| 2.2      | Основные не вычислимые алгоритмы. | Алгоритмы поиска. Последовательный поиск в неупорядоченном массиве. Бинарный поиск. Поиск с барьером.<br>Алгоритмы сортировки. Обменная сортировка методом пузырька. Сортировка выбором. Сортировка вставками. Рекурсивные функции, понятие вычислимой функции.<br>Сложность рекурсивных вычислений. Ханойские башни. |
| <b>1</b> | <b>4 семестр</b>                  |   |