

О. Г. Казанцева

Алгоритмы и структуры данных

**Практикум по дисциплине «Алгоритмы и структуры данных»
для студентов 2 курса специальности Прикладная математика (1-
31 03 03)**

2011

Весь процесс разработки компьютерных программ стандартно разбивается на следующие этапы:

1. **Анализ постановки задачи.** Выяснить при этом входную и выходную информацию, определить идею решения, полноту входной информации, сформулировать, накладываемые на входную информацию ограничения (то есть описать спецификации).

2. **Построение математической модели.** Определить какие математические структуры больше подходят для задачи, существуют ли аналоги решения этой задачи. После чего проверить полноту математической модели, удобство работы с ней, реализации.

3. **Разработка алгоритма.** На этом этапе необходимо тщательно обдумать алгоритм, уяснить его достоинства и недостатки, постараться избавиться от последних и усилить первые.

4. **Анализ алгоритма.** Оценить какой это алгоритм(линейный, с ветвлениями, с циклами). Т.е. оценить его сложность по памяти и/или быстродействию.

5. **Доказательство правильности алгоритма.** Этот этап предусматривает как доказательство конечности алгоритма, так и анализ его работы на всех ветвях на разных типах входных данных.

6. **Реализация алгоритма.** Это означает необходимость написания программы на некотором языке программирования и отладка его на реальном компьютере.

7. **Тестирование программы.** Т.е. провести ее как теоретическое, так и экспериментальное тестирование. Для алгоритма необходимо подготовить полный набор тестов (минимальное подмножество входных данных, покрывающее все случаи)

8. **Оформление документации.** Описать технические характеристики, структуру входных и выходных данных, правила использования программы, при необходимости – реализуемые алгоритмы и возможности модификации.

Требования к выполнению лабораторных работ:

1. При выполнении лабораторной работы необходимо, в соответствии с вариантом, разработать алгоритм, реализовать его в виде объектно-ориентированного приложения на C++ или Java и написать отчет.
2. При работе с программой пользователь должен видеть
 - 1) условие задачи,
 - 2) требования к входным данным,
 - 3) подсказки, в случае некорректного ввода.
3. Разработанная программа должна работать в двух режимах:
 - 1) выполнять алгоритм для одного набора входных данных (в этом случае ввод данных и вывод результата осуществляется в полях формы),
 - 2) выполнять алгоритм для множества наборов входных данных (в этом случае ввод данных осуществляется из файла входных данных, вывод – в файл результатов).

В обоих режимах необходимо оценивать временные затраты на выполнение алгоритма.
4. Отчет по лабораторной работе необходимо представлять в текстовом виде (*.doc), он должен содержать
 - 1) ФИ автора, № группы,
 - 2) условие задачи,
 - 3) описание всех этапов разработки программы.
5. Для седьмого этапа необходимо в виде таблицы предоставить полный набор тестов, результаты и временные затраты.

Лабораторная работа №1

Решение задач методом полного перебора

Задание:

Решите задачу. Оформите решение в соответствии с требованиями к оформлению задач.

При разработке алгоритма предусмотрите возможность сокращения вариантов перебора.

№	Условие
1	<p>№1005 «Уравнение – II». Вводится число N. Написать программу, выводящую все решения уравнения: $X! * N = Y!$. Если решений нет выдать - «NO».</p> <p>Ввод: число N.</p> <p>Вывод: все решения по одному в строке.</p> <p>Пример: Ввод: 12 Вывод: 2 4 11 12</p>
2	<p>№1006 «Число в числе». Вводится число длиной не более 9 цифр. Сколько различных K-разрядных чисел можно составить из цифр данного числа, если любую цифру, входящую в число, можно повторять любое число раз, а K - количество различных цифр в вводимом числе.</p> <p>Ввод: число N.</p> <p>Вывод: ответ.</p> <p>Пример: Ввод: 12121 Вывод: 4</p>
3	<p>№1014 «Юбилей». В связи с открытием олимпиады N человек ($N < 11$) решили устроить вечеринку. Для её проведения достаточно купить MF бутылок фанты, MB бананов, MC тортов. При покупке определенных наборов товаров действуют правила оптовой торговли: стоимость набора товара может отличаться от суммарной стоимости отдельных частей. Написать программу, вычисляющую по исходным данным, минимальный взнос участника вечеринки V.</p> <p>Ввод: 1-я строка - числа N и M ($M \leq 100000$, число возможных наборов), в каждой из следующих M строк по 4 числа: F, B, C, S ($0 \leq F, B, C \leq 1000$, $S \leq 100000$) –соответственно количества бутылок фанты, бананов и тортов в наборе и стоимость набора; в последней строке – числа MF, MB, MC (≤ 9).</p> <p>Вывод: число V.</p>

	<p>Пример: Ввод: 2 3 1 1 1 500 2 2 2 900 2 4 2 1050 2 2 1 Вывод: 450.00</p>
4	<p>№1023 «Расписание». В нулевой момент времени мастеру одновременно поступают N работ. Одновременно может выполняться только одна работа, и если мастер приступил к ее выполнению, то продолжает выполнять ее пока не закончит. Работы пронумерованы от 1 до N ($N < 100$). Для каждой работы задается время ее выполнения - T, кратное целым суткам и C - сумма штрафа, взимаемого за каждые сутки ожидания до начала выполнения работы. Найти такой порядок выполнения работ, чтобы итоговая сумма штрафов по всем работам оказалась минимальной. Вывести значение минимальной суммы штрафов и порядок выполнения работ.</p> <p>Ввод: число N, далее в каждой из N строк: сведения о I-й работе: T_i C_i.</p> <p>Вывод: 1-я строка сумма штрафа. 2-я строка порядок выполнения работ.</p> <p>Пример: Ввод: 3 4 1 1 1 10 3 Вывод: 14 2 3 1</p>
5	<p>№1026 «Точка». На плоскости отмечены все точки, обе координаты которых - натуральные числа. Будет ли видна точка с координатами (n, m) из начала координат - точки (0, 0)? Считаем, что точка видна, если внутри отрезка, соединяющего точки (0,0) и (n, m), нет отмеченных точек. Например, точка (2, 2) не видна из начала координат, так как ее не дает увидеть точка (1, 1). Числа n и m являются натуральными и $n \leq 2000000000$, $m \leq 2000000000$.</p> <p>Ввод: Натуральные числа n, m</p> <p>Вывод: Слово «NO», если точка (n, m) не видна из начала координат, и, на следующей строке, - координаты (x, y) точки, которая не дает увидеть точку (n, m) из начала координат и, расстояние, от которой до начала координат минимальное.</p> <p>Слово «YES», если точка (n, m) видна из начала координат.</p> <p>Пример: Ввод: 100 200 Вывод:</p>

	<p>No 1 2</p>
6	<p>№ 1036 «Строка в строке» Заданы две символьные строки, не содержащие пробелов. Написать программу, вычисляющую количество способов, которыми можно получить вторую строку из первой, вычеркиванием некоторых символов.</p> <p>Ввод: две строки Вывод: число-результат.</p> <p>Пример: Ввод: AAABBBCCSS ABC Вывод: 36</p>
7	<p>№1043 «Восстановление строки». Имеется длинная строка текста S, содержащая N маленьких букв латинского алфавита ($0 < N < 10000$).</p> <p>Выполним над этой строкой циклический сдвиг вправо на один символ, т.е. последний символ строки переместим в первую позицию, а все остальные символы сместим на одну позицию вправо. Таким образом получаем строку S1. Затем выполняем циклический сдвиг вправо для строки S1, получая строку S2 и т.д. После выполнения N сдвигов будем иметь набор строк S_i $i=1..N$ и одной из этих строк будет являться исходная строка S. Упорядочим получившиеся строки в алфавитном порядке, а затем выпишем из каждой строки последний символ и поочередно запишем их в столбец L, т.е. j-ый символ в L - это последний символ j-ой строки в упорядоченном наборе строк. Считаем известной величину K - порядковый номер строки S в упорядоченном наборе строк S_i $i=1..N$. Такое преобразование переупорядочивает символы исходной строки, и делает её легкой для последующего эффективного сжатия простыми алгоритмами.</p> <p>Вам необходимо восстановить исходную строку S после такого преобразования для заданных значений параметров N, K и L.</p> <p>Ввод: Первая строка файла содержит два числа: N - количество символов и K - порядковый номер строки S в упорядоченном наборе строк S_i. Столбец L записан во второй строке файла.</p> <p>Вывод: Полученную исходную строку S вывести в текстовый файл output.txt.</p> <p>Пример: Ввод: 6 2 saraab Вывод: abraca</p>
8	<p>№1063 «Жуки и пауки» На берегу реки сидят жуки и пауки, у которых вместе N ног. У каждого жука по 6 ног, у паука - по 8 ног. Известно, что ног у всех жуков в K раз меньше, чем у всех пауков. Сколько всего жуков и пауков сидит на берегу реки. N и K - натуральные числа. Если решений нет выдать ответ ноль.</p> <p>Ввод: числа N и K (до 100000000). Вывод: одно число – ответ.</p> <p>Пример: Ввод: 360 2</p>

	Вывод: 50
9	<p>№1068 «Найди серединку» Вводятся натуральные числа - А и В ($A < B$), которые не превышают 100000000. Найти число X – середину ряда, состоящего из всех цифр из чисел от А до В записанных подряд, для этого нужно: записать подряд все числа от А до В (включительно), найти в полученном ряду цифру или число из двух цифр (это зависит от четности или нечетности количества цифр в ряду), стоящие ровно посередине ряда. (Пример: для $A=98$, $B=103$, в ряду от 98 до 103 будет записано 16 цифр, средними (8-й и 9-й) цифрами будут 10).</p> <p>Ввод: число К. Вывод: ответ. Пример: Ввод: 98 103 Вывод: 10</p>
10	<p>№1069 «Максимальное четное» Вводится число X, содержащее в своей записи не более 255 знаков. Найти в заданном числе максимальное, натуральное четное число, входящее как подстрока в заданное число. Если четного числа нет – выдать сообщение «NO».</p> <p>Ввод: строка число X. Вывод: одно четное число – ответ, или «NO». Пример: Ввод: 365678.456882 Вывод: 456882</p>
11	<p>№1064 «Периметр» Таблица содержит значения длин N отрезков. Вычислить наибольший периметр треугольника, который можно составить из отрезков из заданной таблицы. Каждый отрезок из таблицы можно использовать в треугольнике один раз. (В треугольнике длина любой из сторон должна быть меньше суммы длин двух других сторон). Если решений нет выдать ответ «0».</p> <p>Ввод: 1-я строка: число N (до 2000). Во 2-й через пробел N значений длин отрезков. Вывод: одно число – ответ. Пример: Ввод: 6 1.8 3.3 6.0 5.0 11.3 5.2 Вывод: 16.2</p>
12	<p>№1065 «Читатель-читататель» Мальчик листает книгу, начиная со страницы с номером N в обратном порядке (в порядке убывания номеров страниц) и подсчитывает при этом сумму нечетных номеров страниц. Вывести получившуюся сумму если известно, что мальчик пролистал K страниц. N и K - натуральные числа ($N \geq K$).</p> <p>Ввод: числа N и K (до 1000000). Вывод: одно число – ответ. Пример:</p>

	<p>Ввод: 362 5 Вывод: 720</p>								
3	<p>1 <u>Задача 54 «Номератор»</u></p> <p>Для печатания номеров пользуются 15-ти разрядным номератором, который печатает по порядку, начиная с заданного начального числа, пятнадцатизначные номера, если число имеет меньше 15-ти разрядов то вместо отсутствующих разрядов печатаются ноли. На печатание каждой цифры расходуется краска, ее расход пропорционален длине ломаной линии, изображающей цифру. Высота цифры в два раза больше ее ширины, цифры имеют следующие формы:</p> <p style="text-align: center;">0123456789</p> <p>Написать программу, которая вычисляла бы минимальный, из заданного интервала (А В), номер, начиная с которого для печати серии из К номеров расходуется минимальное количество краски. $0 \leq A, B < 1\ 000\ 000\ 000\ 000\ 000$; $0 < K \leq 100\ 000$; $A \leq B$; $0 \leq (B-A) \leq 1000\ 000$; границы интервала входят в интервал, а печатаемые номера могут выходить за границы интервала.</p> <p><u>Вход:</u> 1-я строка - число А – начало интервала; 2-я строка - число В - конец интервала; 3-я строка - число К - количество номеров в серии.</p> <p><u>Выход:</u> одно число - результат.</p> <p><u>Пример:</u></p> <table style="width: 100%; border: none;"> <tr> <td style="width: 60%;"><u>Ввод:</u></td> <td style="width: 40%;"><u>Вывод:</u></td> </tr> <tr> <td>987654320123456</td> <td>987654321101001</td> </tr> <tr> <td>987654321123456</td> <td></td> </tr> <tr> <td>78987</td> <td></td> </tr> </table>	<u>Ввод:</u>	<u>Вывод:</u>	987654320123456	987654321101001	987654321123456		78987	
<u>Ввод:</u>	<u>Вывод:</u>								
987654320123456	987654321101001								
987654321123456									
78987									
3	<p>1 <u>Задача 62 «Наибольшее простое»</u></p> <p>Вводится натуральное число, не превосходящее числа 2147 000 000.</p> <p>Написать программу, которая выдавала бы наибольшее простое число, которое возможно получить из заданного числа вычеркиванием цифр. Можно вычеркивать любое количество (и ноль тоже) цифр. Если решения нет выдать - число 0.</p> <p><u>Ввод:</u> М – число.</p> <p><u>Вывод:</u> число - результат.</p> <p><u>Пример:</u></p> <table style="width: 100%; border: none;"> <tr> <td style="width: 60%;"><u>Ввод:</u></td> <td style="width: 40%;"><u>Вывод:</u></td> </tr> <tr> <td>123456</td> <td>23</td> </tr> </table>	<u>Ввод:</u>	<u>Вывод:</u>	123456	23				
<u>Ввод:</u>	<u>Вывод:</u>								
123456	23								
4	<p>1 <u>Задача 37 «Двоичное уравнение»</u></p> <p>Составить программу, решающую уравнение $A * X + B = C$, в котором А,В,С и X - числа, заданные в двоичной системе счисления. Ввод корректен.</p> <p><u>Ввод:</u> Три строки: три числа А В С , длиной до 30 знаков.</p> <p><u>Вывод:</u> Единственное число - X – решение уравнения в двоичной системе счисления, округленное до двух знаков после запятой.</p>								

	<u>Пример :</u>		
	<u>Ввод:</u>		<u>Вывод:</u>
	111	101.00	
	10001	110100	

Репозиторий ВГУ

Лабораторная работа № 2

Решение задач методом динамического программирования

Задание:

Решите задачу. Оформите решение в соответствии с требованиями к оформлению задач.

№	Условие		
1	<p><u>Задача 9 «Шифр»</u></p> <p>Заданы три натуральных числа: два исходных числа и номер члена, который является ключом к шифру. Каждый последующий член последовательности получается путем прибавления к сумме двух предыдущих членов, суммы цифр суммы двух предыдущих членов. Написать программу, которая определяла бы ключ к шифру. Исходные данные - натуральные числа до 1000. Ввод корректен.</p> <p><u>Ввод:</u> C1 C2 N - разделенные пробелом исходные числа и номер искомого члена.</p> <p><u>Вывод:</u> число-результат.</p> <p><u>Пример :</u></p> <table style="width: 100%; border: none;"> <tr> <td style="text-align: left;"><u>Ввод:</u> 39 11 5</td> <td style="text-align: right;"><u>Вывод:</u> 140</td> </tr> </table>	<u>Ввод:</u> 39 11 5	<u>Вывод:</u> 140
<u>Ввод:</u> 39 11 5	<u>Вывод:</u> 140		
2	<p><u>Задача 10 «Ход конем»</u></p> <p>На клетчатой доске размером $N*N$ клеток в клетке с координатами A, B (координаты отсчитывать от левого нижнего угла), стоит шахматный конь. Конь за один ход перемещается одновременно на две клетки вперед и одну в сторону в любом горизонтальном или вертикальном направлении в пределах заданной доски. Написать программу, которая вычисляла бы количество клеток, в которые конь не сможет попасть за M и менее M ходов. Ввод корректен, M и N: $1 < N < 101$, $0 \leq M < 1001$.</p> <p><u>Ввод:</u> N A B M - : размер доски, координаты коня и количество ходов.</p> <p><u>Вывод:</u> число - результат.</p> <p><u>Пример:</u></p> <table style="width: 100%; border: none;"> <tr> <td style="text-align: left;"><u>Ввод:</u> 5 1 1 3</td> <td style="text-align: right;"><u>Вывод:</u> 3</td> </tr> </table>	<u>Ввод:</u> 5 1 1 3	<u>Вывод:</u> 3
<u>Ввод:</u> 5 1 1 3	<u>Вывод:</u> 3		
3	<p><u>Задача 15 «Разноделящиеся числа»</u></p> <p>Будем называть два целых числа «разноделящимися», если большее по модулю делится без остатка на меньшее по модулю и нет ни</p>		

	<p>одной такой цифры, которая входила бы в десятичную запись обоих этих чисел. Меньшее по модулю из этих чисел будем называть «разноделящимся» делителем большего по модулю числа. Составить программу находящую на заданном интервале число (максимальное) имеющее наибольшее количество различных «разноделящихся» делителей. Вывести это число и количество его различных «разноделящихся» делителей. В отдельно взятом одном из «разноделящихся» чисел цифры могут повторяться; А и В - границы интервала числа, не превосходящие по модулю число 2 000 000; границы входят в интервал, задаются и вводятся в произвольном порядке; длина интервала не превосходит 202; если на заданном интервале несколько чисел имеют равное наибольшее количество различных «разноделящихся» делителей, то взять максимальное из них; число «1» считать делителем (обычным) любого числа; знак числа цифрой не считать;</p> <p>Ввод: два числа: А и В - границы интервала.</p> <p>Вывод: два числа: первое - на заданном интервале число, имеющее наибольшее количество различных «разноделящихся» делителей, второе - количество различных «разноделящихся» делителей первого числа.</p> <p>Пример :</p> <table data-bbox="437 1061 1289 1137"> <tr> <td>Ввод:</td> <td>Вывод:</td> </tr> <tr> <td>9 16</td> <td>16 6</td> </tr> </table>	Ввод:	Вывод:	9 16	16 6
Ввод:	Вывод:				
9 16	16 6				
4	<p>Задача 14 «Хакер»</p> <p>Хакеру при помощи подслушивающего устройства удалось подслушать пароль доступа к секретной компьютерной базе данных, который был продиктован по телефону посимвольно (в звуковой форме наименование букв и цифр, например: буква М произносилась как 'эм', а записывалась хакером в виде одного символа). Хакер записал пароль в виде строки, состоящей из заглавных букв и цифр. Но радость хакера была преждевременной, так как по телефону ни слова не было сказано большие или маленькие, русские или латинские буквы используются в пароле, а система защиты базы данных различает буквы по этим признакам, кроме этого, записывая буквы, в спешке, хакер записывал услышанные знаки, не учитывая, что некоторые из них в различных алфавитах пишутся одинаково, но звучат по-разному (например: Р "эр"(рус.) и "пэ" (лат.)). Составить программу, подсчитывающую за какое число попыток ввода пароля, хакеру гарантированно удастся проникнуть в интересующую его базу данных. Ввод корректен. В пароле могут использоваться латинские и русские буквы и цифры, других символов в пароле нет. Длина пароля не превышает 80 символов. Уточнение: одинаково произносиосимыми в различных алфавитах считать буквы:</p>				

	<p>русские - АБВД ЕЖИКЛМНОПРСТУФЦ латинские - ABVDE G I KLMNO PRSTU FC одинаковыми по написанию в различных алфавитах считать буквы:</p> <p>русские - АВЕКМНОРСТХ латинские - АВЕКМНОРСТХ.</p> <p>Ввод: строка-пароль, записанная цифрами и заглавными буквами. Вывод: число - результат. Пример: Ввод: АВ12 Вывод: 32</p>
5	<p>Задача 18 «Визы» Имеется файл из $N \leq 1000$ записей, в каждой из которых содержатся дата открытия и дата закрытия визы в формате: ГТТГ.ММ.ДД где: ГТТГ - год(4-ре цифры), ММ - № месяца (2 цифры), ДД - число (2 цифры). День открытия и день закрытия визы входят в срок действия визы. Составить программу, вычисляющую по имеющемуся файлу, какое было максимальное число открытых одновременно виз. Ввод корректен.</p> <p>Ввод: 1-я строка: число N - количество записей, далее N - строк: в каждой сначала дата открытия, затем дата закрытия визы, разделенные одним пробелом. Вывод: число - результат. Пример : Ввод: 4 2000.01.01 2000.05.11 1997.11.30 1999.12.31 1998.06.07 2000.10.20 1999.01.012000.01.01 Вывод: 3</p>
6	<p>Задача 43 «Сказка» Иван Царевич сражается со Змеем Горынычем, у которого N ($N < 30333$) голов, которые Иван отрубает так: если у змея четное число голов - он отрубает 9 голов, а если нечетное - 11 голов одним ударом. После каждого удара, если у Змея осталась не срубленной хотя бы одна голова, то отрастает еще 8 новых голов. Змей побежден, если после очередного удара мечом у него не осталось ни одной головы. Написать программу, которая вычисляла бы, какое количество ударов понадобится Ивану Царевичу для победы над Змеем Горынычем, и сколько при этом он отрубит голов у Змея. Вход: N - начальное число голов у Змея Горыныча. Выход: Разделенные пробелом два числа: первое – число ударов,</p>

	<p>второе - число срубленных при этом голов.</p> <p><u>Пример :</u></p> <p><u>Ввод:</u> 33</p> <p><u>Вывод:</u> 13 129</p>
7	<p><u>Задача 44 «Гномы»</u></p> <p>Семь гномов имеют каждый по одной кружке, емкостью от 100 до 1000 грамм (емкость кружки - вещественное число из указанного диапазона, заданное в формате с шестью знаками после запятой). Для приготовления своих волшебных смесей гномы используют живую и мертвую воду, источники которой находятся возле их жилищ. Воду из источников брать и выливать обратно можно в неограниченных количествах, но приготовление волшебных смесей требует очень большой точности (до долей миллиграммов, или шести знаков после запятой). Написать программу, которая по заданным семи емкостям кружек, определяла с точностью до шести знаков после запятой, минимальный объем воды, который можно точно отмерить данным набором кружек.</p> <p><u>Вход:</u> в строке 7 чисел, разделенных пробелами - емкости кружек.</p> <p><u>Выход:</u> Число – решение в формате вида: #####.#####.</p> <p><u>Пример: Вход:</u> 100.000000 125.125000 250.125000 250.250000 500.000000 750.125000 1000.000000</p> <p><u>Выход:</u> 0.125000</p>
8	<p><u>Задача 32 «Взвешивания»</u></p> <p>Определить, можно ли на чашечных весах взвесить груз весом G при помощи набора из N гирь массами M_1, M_2, \dots, M_N. Числа G, N, M_i - целые; ответ выдать в форме YES - если взвесить можно, NO - если нельзя.</p> <p><u>Ввод:</u> 1-я строка $G N$ - вес груза и количество гирь, 2-я строка через пробелы массы гирь M_i.</p> <p><u>Вывод:</u> ответ YES или NO .</p> <p><u>Пример :</u></p> <p><u>Ввод:</u> 150 4 100 30 300 20</p> <p><u>Вывод:</u> YES</p>
9	<p><u>Задача 47 «Контрольные суммы»</u></p> <p>При зашифровке текста для каждого слова в тексте</p>

	<p>вычисляется его контрольная сумма, равная сумме порядковых номеров в алфавите букв из которых состоит слово, например: для слова АВba контрольная сумма равна $(1+2+2+1)=6$. Написать программу, которая вычисляла бы количество различных отличных от нулевой контрольных сумм, встречающихся в заданном тексте. В исходном тексте до 25000 слов, слова разделены одним или несколькими пробелами, длина слова до 80 символов, для большой и малой букв порядковые номера в алфавите считать одинаковыми, в контрольную сумму слова считать только номера латинских букв, для других символов номер в алфавите считать равным нулю.</p> <p>Вход: Исходный текст. Выход: число - результат. Пример: Ввод: Good BYE, MY LoVe! Вывод: 4</p>
<p>1 0</p>	<p>Задача 39 «Строки»</p> <p>Вводятся две строки. Составить программу, отвечающую: 1) можно ли получить вторую строку выбрасыванием символов из первой строки читаемой обычно (слева направо). 2) можно ли получить вторую строку выбрасыванием символов из первой строки читаемой справа налево. Ответы выдать словами: YES - если можно получить и NO - если нельзя, Слова YES и NO должны быть выведены большими латинскими буквами.</p> <p>Ввод: 1-я строка: из которой выбрасыванием символов, получают вторую. 2-я строка: та которую нужно получить.</p> <p>Вывод: YES - если можно получить из 1-й строки 2-ю и NO - если нельзя, 1-я слово: можно ли получить строку 2 выбрасыванием символов из 1-й строки читаемой обычно, а 2-е слово: то же, но для 1-й строки, читаемой наоборот.</p> <p>Пример: Ввод: ИНФОРМАТИКА НОРМА Вывод: YES NO</p>
<p>1 1</p>	<p>Задача 8 «Разные отрезки»</p> <p>Отрезок на плоскости задан координатами его концов. Написать программу, которая вычисляла бы количество различных отрезков, целиком лежащих на данном отрезке, обе координаты концов, которых - целочисленные. Концы принадлежат отрезкам.</p> <p>Ввод: X1 Y1 X2 Y2 - разделенные пробелами координаты концов отрезка. Вывод: число - результат.</p>

	<p><u>Пример :</u></p> <p><u>Ввод:</u> 0.5 0.5 -3.3 -3.3</p> <p><u>Вывод:</u> 6</p>
1 2	<p><u>Задача 62 «Наибольшее простое»</u></p> <p>Вводится натуральное число, не превосходящее числа 2147 000 000.</p> <p>Написать программу, которая выдавала бы наибольшее простое число, которое возможно получить из заданного числа вычеркиванием цифр. Можно вычеркивать любое количество (и ноль тоже) цифр. Если решения нет выдать - число 0.</p> <p><u>Ввод:</u> M – число. <u>Вывод:</u> число - результат.</p> <p><u>Пример:</u></p> <p><u>Ввод:</u> 123456</p> <p><u>Вывод:</u> 23</p>
1 3	<p><u>Задача 53 «Ярмарка»</u></p> <p>Между населенными пунктами Дубово и Клиново имеется железнодорожная ветка длиной S километров и с N остановочными пунктами, пронумерованными по порядку возрастания номеров по пути следования из Дубово в Клиново. Приближается ярмарка. Организаторы желают провести ее в таком месте, чтобы сумма расстояний, которые преодолели по пути на ярмарку все ее участники, была наименьшей. О каждом остановочном пункте имеются следующие сведения: R_i -расстояние от i-го пункта до Клиново (по железной дороге), K_i -количество участников ярмарки, проживающих в i-м пункте. Из Дубово на ярмарку поедет Z , а из Клиново - Y человек. Написать программу, выводящую номер пункта, в котором следует проводить ярмарку (если ярмарка в Дубово - вывести 0 , если в Клиново - вывести число N+1, если возможны несколько пунктов - вывести их номера в порядке возрастания). Ввод корректен. $0 < N < 1001$, $0 < R_i, S < 9001$, $0 \leq K_i, X, Y \leq 100\ 000$.</p> <p><u>Вход:</u> N+1 строка: в 1-й строке : N- число остановок, S - расстояние между селениями, Z, Y - число участников ярмарки из Дубово и Клиново, в каждой из следующих N строк: R_i, K_i- сведения о i-м пункте.</p> <p><u>Выход:</u> номер или номера (в порядке возрастания) пункта(ов), где следует проводить ярмарку.</p> <p><u>Пример:</u></p> <p><u>Ввод:</u> 3 13 3 1 12 2</p> <p><u>Вывод:</u> 2</p>

	<p>8 2 1 3</p>
1 4	<p><u>Задача 27 «Таблица»</u></p> <p>В двумерной таблице размером N на N элементов встречается ровно N различных чисел, причем в любой строке все элементы различные. Подсчитать, на сколько полей заканчивается число возможных различных значений этой таблицы. Ввод корректен. $0 < N < 1\ 000\ 000\ 000$.</p> <p><u>Ввод:</u> число – N.</p> <p><u>Вывод:</u> число-результат - на сколько полей заканчивается число возможных различных значений таблицы, число представляется в виде последовательности цифр в десятичной записи (другие формы представления числа - недопустимы).</p> <p><u>Пример :</u></p> <p><u>Ввод:</u> 6</p> <p><u>Вывод:</u> 6</p>
1 5	<p><u>Задача 12 «Параллелоландия»</u></p> <p>Страна Параллелоландия имеет форму прямоугольного параллелепипеда размера A на B на C, состоящего из кубиков с единичной длиной ребра. Гражданами этой страны являются все прямоугольные параллелепипеды, расположенные в пределах ее границ и состоящие из целого числа единичных кубиков. Родственниками в этой стране считаются те параллелепипеды, которые имеют хотя бы один общий единичный кубик. Еще в Параллелоландии есть общество кубов, членами которого являются граждане, имеющие все три одинаковых размера. Необходимо написать программу, которая поможет одному маленькому кубику, расположенному по адресу A1, B1, C1 (число A1 - это номер ряда, в размере A страны, в котором расположен этот единичный кубик, аналогично числа B1 и C1), подсчитать сколько в Параллелоландии граждан не являющихся членами общества кубов не являются родственниками маленького куба. A,B,C,A1,B1,C1 - натуральные числа не превосходящие 50..</p> <p><u>Ввод:</u> A B C - разделенные пробелами - размеры страны, в первой строке и –</p> <p>A1 B1 C1 - три числа, разделенные пробелами - адрес кубика - во второй строке.</p> <p><u>Вывод:</u> число - результат.</p> <p><u>Пример:</u></p> <p><u>Ввод:</u> 2 1 2 1 1 2</p> <p><u>Вывод:</u> 2</p>

<p>1 6</p>	<p><u>Задача 67 «Тир»</u> Стрелок приходит в тир и ставит в банк сумму X. Затем он стреляет, за точный выстрел его сумму увеличивают на P %, а за неточный уменьшают на P%. Написать программу, которая вычисляла количество различных возможных вариантов стрельбы, при которых у игрока окажется сумма равная X1. Проценты начисляются и снимаются с той суммы, которая на счету у стрелка в момент произведения выстрела. Стрелок делает не более 20 выстрелов. <u>Ввод:</u> X X1 P. <u>Вывод:</u> число - ответ. <u>Пример:</u> <u>Ввод:</u> 1000 10 980.1 <u>Вывод:</u> 6 №1050 «Спички». Для заданного количества спичек необходимо подсчитать количество различных всех возможных тождеств вида: A+B=C или A-B=C, где A,B,C числа в римской записи от 1 до 89, используя для каждого тождества количество спичек не более заданного. Числа в римской записи создаются по правилам: 1) I – 1 V – 5 X – 10 L – 50; 2) количество символов должно быть минимально; 3) из большей можно вычесть только одну меньшую по величине цифру; Для обозначения символа “I” и знака “-“ расходуется по одной спичке, для остальных цифр и знаков – по две. <u>Ввод:</u> число спичек. <u>Вывод:</u> количество верных тождеств. <u>Пример:</u> Ввод: 7 Вывод: 1</p>
<p>1 7</p>	<p><u>Задача 70 «Отрезание квадратов»</u> От прямоугольного листа бумаги с целочисленными длинами сторон отрезают квадрат максимально возможного размера, если оставшаяся часть прямоугольник, а не квадрат, то с ней делают то же, что и с исходным прямоугольником, так повторяют пока оставшаяся часть не станет квадратом. Составить алгоритм, выводящий минимальные длины сторон прямоугольника, при разрезании которого получится ровно N различных по величине квадратов. N не превосходит 1000. <u>Ввод:</u> N – число различных квадратов. <u>Вывод:</u> минимальные ширина и длина листа.</p>

	<p><u>Пример :</u></p> <p><u>Ввод:</u> 10</p> <p><u>Вывод:</u> 89 144</p>
1 8	<p><u>Задача 66 «Четыре цифры»</u></p> <p>Вводятся четыре произвольные десятичные цифры. Написать программу, которая находит и выводит максимальное возможное число, которое можно записать в математической форме записи при помощи введенных цифр. В математической форме записи операция возведение в степень не имеет значка, а обозначается записью чисел-показателей степени в виде верхних индексов. Программа должна выдать число в математической форме записи в виде строки – арифметического выражения (не вычисляя значение числа), для операции возведение в степень используя значок: ^ . В случае введенных четырех нулей, достаточно выдать просто один нуль.</p> <p><u>Ввод:</u> A1 A2 A3 A4 - четыре цифры.</p> <p><u>Вывод:</u> число – ответ в описанной в условии форме записи.</p> <p><u>Пример:</u></p> <p><u>Ввод:</u> 1 2 0 0</p> <p><u>Вывод:</u> 2^100</p>
1 9	<p><u>Задача 15 «Разноделящиеся числа»</u></p> <p>Будем называть два целых числа «разноделящимися», если большее по модулю делится без остатка на меньшее по модулю и нет ни одной такой цифры, которая входила бы в десятичную запись обоих этих чисел. Меньшее по модулю из этих чисел будем называть «разноделящимся» делителем большего по модулю числа. Составить программу находящую на заданном интервале число (максимальное) имеющее наибольшее количество различных «разноделящихся» делителей. Вывести это число и количество его различных «разноделящихся» делителей. В отдельно взятом одном из «разноделящихся» чисел цифры могут повторяться; A и B - границы интервала числа, не превосходящие по модулю число 2 000 000; границы входят в интервал, задаются и вводятся в произвольном порядке; длина интервала не превосходит 202; если на заданном интервале несколько чисел имеют равное наибольшее количество различных «разноделящихся» делителей, то взять максимальное из них; число «1» считать делителем (обычным) любого числа; знак числа цифрой не считать;</p> <p><u>Ввод:</u> два числа: A и B - границы интервала.</p> <p><u>Вывод:</u> два числа: первое - на заданном интервале число, имеющее наибольшее количество различных «разноделящихся» делителей, второе -</p>

Лабораторная № 3

Фундаментальные структуры данных.

Выполните упражнения (используя объектно-ориентированное программирование), сформулированные на лекциях.

Упр.1: реализовать базовые операции для массива.

Упр.2: реализовать базовые операции для одно- и двунаправленного списка, реализованного через массивы.

Упр.3: реализовать базовые операции для одно- и двунаправленного списка, реализованного через последовательности связанных компонент .

Упр.4: реализовать операции конкатенации и расщепления одно- и двунаправленного списка, реализованного через массивы и через последовательности связанных компонент .

Упр.5: реализовать операции добавления и удаления элемента в стеке, реализованном через массивы и через последовательности связанных компонент .

Упр.6: реализовать операции добавления и удаления элемента в обычной и в циклической очереди, реализованном через массивы и через последовательности связанных компонент.

Упр.7: реализовать операции добавления и удаления элемента в дек (и в начало и в конец очереди), реализованном через массивы и через последовательности связанных компонент.

Лабораторная работа 4

«Основные алгоритмы внутренней и внешней сортировки»

Задание. Разработайте информационную систему в соответствии с вариантом задания (*вариант задания и вариант алгоритма сортировки определяет преподаватель*). Реализуйте функции для ввода, вывода данных (из внешних источников) и функцию упорядочения по различным признакам.

Для реализации упорядочения используйте один из алгоритмов сортировки, *указанный преподавателем*:

- а) сортировка включение (прямое, двоичное),
- б) сортировка выбором,
- в) пузырьковая сортировка,
- г) шейкерная сортировка,
- д) сортировка слиянием
- е) быстрая сортировка,
- ж) лексикографическая сортировка,
- з) внешняя сортировка

Варианты задания

1. Экзаменационная ВЕДОМОСТЬ по некоторой дисциплине: фамилия И.О. СТУДЕНТА, номер студенческого билета, оценка.
2. РАСПИСАНИЕ отправления пригородных ПОЕЗДОВ по некоторой станции: номер поезда, конечная станция, время отправления, время прибытия на конечную станцию.
3. Автобусный ПАРК: гос.номер АВТОБУСА, модель, год выпуска, количество мест для сидения, количество мест для стояния.
4. Аптечный СКЛАД: наименование ЛЕКАРСТВА, количество упаковок, количество таблеток в упаковке, масса таблетки.
5. Книжный МАГАЗИН: автор КНИГИ, название, год издания, цена, количество экземпляров.
6. Ящик электронной ПОЧТЫ: отправитель СООБЩЕНИЯ, текст сообщения, даты отправления.
7. КАТАЛОГ музыкальных ПРОИЗВЕДЕНИЙ: имя исполнителя, название, стиль, продолжительность.
8. База данных ОТДЕЛА КАДРОВ: Фамилия И.О. СОТРУДНИКА, пол, специальность, год рождения, стаж работы.
9. База данных ГАИ: гос.номер АВТОМОБИЛЯ, марка, модель, Фамилия И.О. владельца, год выпуска.
10. РАСПИСАНИЕ ЗАНЯТИЙ: день недели, номер пары, название дисциплины, Фамилия И.О. преподавателя.

Лабораторная работа № 5

Тема: Основные алгоритмы на графах.

1. С помощью поиска в глубину в графе выделить связанные компоненты неориентированного графа.
2. С помощью поиска в глубину в графе определить является ли граф двудольным.
3. С помощью поиска в ширину в графе выделить связанные компоненты неориентированного графа.
4. С помощью поиска в ширину в графе определить является ли граф двудольным.
5. С помощью поиска в ширину реализовать алгоритм поиска наименьшего (по числу ребер) пути из заданной вершины до всех оставшихся.
6. С помощью поиска в глубину в графе выделить связанные компоненты неориентированного графа.
7. С помощью поиска в глубину в графе определить является ли граф двудольным.
8. С помощью поиска в глубину реализовать алгоритм поиска наименьшего (по числу ребер) пути из заданной вершины до всех оставшихся.
9. С помощью поиска в ширину в графе выделить связанные компоненты неориентированного графа.
10. С помощью поиска в ширину реализовать алгоритм поиска наименьшего (по числу ребер) пути из заданной вершины до всех оставшихся.
11. С помощью поиска в ширину в графе определить является ли граф двудольным.
12. С помощью поиска в глубину реализовать алгоритм поиска наименьшего (по числу ребер) пути из заданной вершины до всех оставшихся.

Лабораторная работа 6. Структуры данных: кучи, деревья.

1. Создать программу, позволяющую заполнять и отображать генеалогическое дерево.
2. Создайте и нарисуйте полное бинарное дерево.
3. Используя структуру данных d-куча найдите k-й минимальный элемент последовательности.
4. В файловой системе каталог файлов организован в виде упорядоченного бинарного дерева. Каждый узел обозначает файл и содержит имя файла, а также среди прочего дату последнего обращения к нему, закодированную в виде целого числа. Напишите программу, которая обходит дерево и удаляет все файлы, последнее обращение к которым происходило до определенной даты.
5. Построить словарь с использованием структуры данных: бинарное поисковое дерево. Предусмотреть следующие операции: добавление слова, поиск, удаление.
6. -7. Создать словарь, используя структуру данных: красно-черное дерево. Предусмотреть следующие операции: добавление слова, поиск, удаление.
8. -9. Создать словарь, используя структуру данных: AVL- дерево. Предусмотреть следующие операции: добавление слова, поиск, удаление.
10. Для заданного текста (ввод из файла) построить частотный словарь, используя структуру данных бинарное поисковое дерево.
11. Используя структуру данных d-куча выполните сортировку последовательности.