***Задание к лабораторной работе №1***

*В приведенных ниже заданиях необходимо вывести внизу фамилию разработчика, дату получения задания, а также дату сдачи задания. Для работы с динамическим массивом вводить его с клавиатуры и необходимо использовать утилитный класс Arrays и его стандартные методы. В заданиях на числа объект можно создавать в виде массива символов. Ввести n чисел с консоли (1-15 вариант добавлять новый элемент в начало списка, 16-30 – в конец списка).*

1. Найти самое короткое и самое длинное число. Вывести найденные числа и их длину.

2. Упорядочить и вывести числа в порядке возрастания (убывания) значений их длины.

3. Вывести на консоль те числа, длина которых меньше (больше) средней, а также длину. 4. Найти число, в котором число различных цифр минимально. Если таких чисел несколько, найти первое из них.

5. Найти количество чисел, содержащих только четные цифры, а среди них — количество чисел с равным числом четных и нечетных цифр.

6. Найти число, цифры в котором идут в строгом порядке возрастания. Если таких чисел несколько, найти первое из них.

7. Найти число, состоящее только из различных цифр. Если таких чисел несколько, найти первое из них.

8. Среди чисел найти число-палиндром. Если таких чисел больше одного, найти второе.

9. Найти корни квадратного уравнения. Параметры уравнения передавать с командной строкой.

10. Вывести элементы массива в обратном порядке.

11. Определить принадлежность некоторого значения k интервалам (n, m], [n, m), (n, m), [n, m].

12. Вывести на экран все числа от 1 до 100, которые делятся на 3 без остатка.

13. Написать код программы, которая бы переводила числа из десятичной системы счисления в любую другую.

14. Написать код программы, которая бы переводила числа одной любой системы счисления в любую другую.

15. Ввести число от 1 до 12. Вывести на консоль название месяца, соответствующего данному числу. Осуществить проверку корректности ввода чисел.

16. Ввести с консоли n-размерность матрицы a[n][n]. Задать значения элементов матрицы в интервале значений от -n до n с помощью генератора случайных чисел.

17. Упорядочить строки (столбцы) матрицы в порядке возрастания значений элементов k-го столбца (строки).

18. Выполнить циклический сдвиг заданной матрицы на k позиций вправо (влево, вверх, вниз).

19. Найти и вывести наибольшее число возрастающих\убывающих элементов матрицы, идущих подряд.

20. Найти сумму элементов матрицы, расположенных между первым и вторым положительными элементами каждой строки.

21. Вывести числа от 1 до k в виде матрицы N x N слева направо и сверху вниз.

22. Округлить все элементы матрицы до целого числа.

23. Повернуть матрицу на 90, 180 или 270 градусов против часовой стрелки.

24. Вычислить определитель матрицы.

25. Построить матрицу, вычитая из элементов каждой строки матрицы ее среднее арифметическое.

26. Найти максимальный элемент(ы) в матрице и удалить из матрицы все строки и столбцы, его содержащие.

27. Уплотнить матрицу, удаляя из нее строки и столбцы, заполненные нулями.

28. В матрице найти минимальный элемент и переместить его на место заданного элемента путем перестановки строк и столбцов.

29. Преобразовать строки матрицы таким образом, чтобы элементы, равные нулю, располагались после всех остальных.

30. Перестроить матрицу, переставляя в ней строки так, чтобы сумма элементов в строках полученной матрицы возрастала.

***Задание к лабораторной работе №2***

*Создать классы, спецификации которых приведены ниже. Определить конструкторы и методы setТип(), getТип(), toString(). Определить дополнительно методы в классе, создающем массив объектов. Реализовать методы сложения, вычитания, умножения и деления объектов (для тех классов, объекты которых могут поддерживать арифметические действия). Задать критерий выбора данных и вывести эти данные на консоль. В каждом классе, обладающем информацией, должно быть объявлено несколько конструкторов.*

1. Student: id, Фамилия, Имя, Отчество, Дата рождения, Адрес, Телефон, Факультет, Курс, Группа. Создать массив объектов. Вывести: a) список студентов заданного факультета; b) списки студентов для каждого факультета и курса; c) список студентов, родившихся после заданного года; d) список учебной группы.

2. Customer: id, Фамилия, Имя, Отчество, Адрес, Номер кредитной карточки, Номер банковского счета. Создать массив объектов. Вывести: a) список покупателей в алфавитном порядке; b) список покупателей, у которых номер кредитной карточки находится в заданном интервале.

3. Patient: id, Фамилия, Имя, Отчество, Адрес, Телефон, Номер медицинской карты, Диагноз. Создать массив объектов. Вывести: a) список пациентов, имеющих данный диагноз; b) список пациентов, номер медицинской карты которых находится в заданном интервале.

4. Abiturient: id, Фамилия, Имя, Отчество, Адрес, Телефон, Оценки. Создать массив объектов. Вывести: a) список абитуриентов, имеющих неудовлетворительные оценки; b) список абитуриентов, у которых сумма баллов выше заданной; c) выбрать заданное число n абитуриентов, имеющих самую высокую сумму баллов (вывести также полный список абитуриентов, имеющих полупроходную сумму).

5. Book: id, Название, Автор(ы), Издательство, Год издания, Количество страниц, Цена, Тип переплета. Создать массив объектов. Вывести: a) список книг заданного автора; b) список книг, выпущенных заданным издательством; c) список книг, выпущенных после заданного года.

6. House: id, Номер квартиры, Площадь, Этаж, Количество комнат, Улица, Тип здания, Срок эксплуатации. Создать массив объектов. Вывести: a) список квартир, имеющих заданное число комнат; b) список квартир, имеющих заданное число комнат и расположенных на этаже, который находится в заданном промежутке; c) список квартир, имеющих площадь, превосходящую заданную.

7. Phone: id, Фамилия, Имя, Отчество, Адрес, Номер кредитной карточки, Дебет, Кредит, Время городских и междугородных разговоров. Создать массив объектов. Вывести: a) сведения об абонентах, у которых время внутригородских разговоров превышает заданное; b) сведения об абонентах, которые пользовались междугородной связью; c) сведения об абонентах в алфавитном порядке.

8. Car: id, Марка, Модель, Год выпуска, Цвет, Цена, Регистрационный номер. Создать массив объектов. Вывести: a) список автомобилей заданной марки; b) список автомобилей заданной модели, которые эксплуатируются больше n лет; c) список автомобилей заданного года выпуска, цена которых больше указанной.

9. Product: id, Наименование, UPC, Производитель, Цена, Срок хранения, Количество. Создать массив объектов. Вывести: a) список товаров для заданного наименования; b) список товаров для заданного наименования, цена которых не превосходит заданную; c) список товаров, срок хранения которых больше заданного.

10.Train: Пункт назначения, Номер поезда, Время отправления, Число мест (общих, купе, плацкарт, люкс). Создать массив объектов. Вывести: a) список поездов, следующих до заданного пункта назначения; b) список поездов, следующих до заданного пункта назначения и отправляющихся после заданного часа; c) список поездов, отправляющихся до заданного пункта назначения и имеющих общие места.

11.Bus: Фамилия и инициалы водителя, Номер автобуса, Номер маршрута, Марка, Год начала эксплуатации, Пробег. Создать массив объектов. Вывести: a) список автобусов для заданного номера маршрута; b) список автобусов, которые эксплуатируются больше заданного срока; c) список автобусов, пробег у которых больше заданного расстояния.

12.Airline: Пункт назначения, Номер рейса, Тип самолета, Время вылета, Дни недели. Создать массив объектов. Вывести: a) список рейсов для заданного пункта назначения; b) список рейсов для заданного дня недели; c) список рейсов для заданного дня недели, время вылета для которых больше заданного.

13. Определить класс Дробь (Рациональная Дробь) в виде пары чисел m и n. Объявить и инициализировать массив из k дробей, ввести/вывести значения для массива дробей. Создать массив/список/множество объектов и передать его в метод, который изменяет каждый элемент массива с четным индексом путем добавления следующего за ним элемента.

14. Определить класс Квадратное уравнение. Реализовать методы для поиска корней, экстремумов, а также интервалов убывания/возрастания. Создать массив/список/множество объектов и определить наибольшие и наименьшие по значению корни.

15. Определить класс Интервал с учетом включения/невключения концов. Создать методы по определению пересечения и объединения интервалов, причем интервалы, не имеющие общих точек, пересекаться/объединятся не могут. Объявить массив/список/множество и n интервалов и определить расстояние между самыми удаленными концами.

16. Определить класс Точка на плоскости (в пространстве) и во времени. Задать движение точки в определенном направлении. Создать методы по определению скорости и ускорения точки. Проверить для двух точек возможность пересечения траекторий. Определить расстояние между двумя точками в заданный момент времени.

17. Определить класс Треугольник на плоскости. Определить площадь и периметр треугольника. Создать массив/список/множество объектов и подсчитать количество треугольников разного типа (равносторонний, равнобедренный, прямоугольный, произвольный). Определить для каждой группы наибольший и наименьший по площади (периметру) объект.

18. Определить класс Четырехугольник на плоскости. Определить площадь и периметр четырехугольника. Создать массив/список/множество объектов и подсчитать количество четырехугольников разного типа (квадрат, прямоугольник, ромб, произвольный). Определить для каждой группы наибольший и наименьший по площади (периметру) объект.

19. Определить класс Окружность на плоскости. Определить площадь и периметр. Создать массив/список/множество объектов и определить группы окружностей, центры которых лежат на одной прямой. Определить наибольший и наименьший по площади (периметру) объект.

20. Определить класс Прямая на плоскости (пространстве). Определить точки пересечения прямой с осями координат. Определить координаты пересечения двух прямых. Создать массив/список/множество объектов и определить группы параллельных прямых.

21. Определить класс Прямая на плоскости (в пространстве), параметры которой задаются с помощью Рациональной Дроби. Определить точки пересечения прямой с осями координат. Определить координаты пересечения двух прямых. Создать массив/список/множество объектов и определить группы параллельных прямых.

22. Определить класс Окружность на плоскости, координаты центра которой задаются с помощью Рациональной Дроби. Определить площадь и периметр. Создать массив/список/множество объектов и определить группы окружностей, центры которых лежат на одной прямой. Определить наибольший и наименьший по площади (периметру) объект.

23. Определить класс Точка в пространстве, координаты которой задаются с помощью Рациональной Дроби. Создать методы по определению расстояния между точками и расстояния до начала координат. Проверить для трех точек возможность нахождения на одной прямой.

24.. Определить класс Четырехугольник на плоскости, вершины которого имеют тип Точка. Определить площадь и периметр четырехугольника. Создать массив/список/множество объектов и подсчитать количество четырехугольников разного типа (квадрат, прямоугольник, ромб, произвольный). Определить для каждой группы наибольший и наименьший по площади (периметру) объект.

25. Определить класс Вектор. Реализовать методы инкремента, декремента, индексирования. Определить массив из m объектов. Каждую из пар векторов передать в методы, возвращающие их скалярное произведение и длины. Вычислить и вывести углы между векторами.

26. Определить класс Вектор. Реализовать методы для вычисления модуля вектора, скалярного произведения, сложения, вычитания, умножения на константу. Объявить массив объектов. Написать метод, который для заданной пары векторов будет определять, являются ли они коллинеарными или ортогональными.

27. Passport: идентификационный номер, Фамилия, Имя, Отчество, номер, Дата рождения, срок действия. Создать массив объектов. Вывести: a) поиск по идентификационному номеру; b) срок действия меньше 3-х месяцев; c) дата рождения, заданная интервалом – вывести все записи.

28. Определить класс Полином c коэффициентами типа Рациональная Дробь. Объявить массив/список/множество из n полиномов и определить сумму полиномов массива.

29. Определить класс Булева матрица (BoolMatrix). Реализовать методы для логического сложения (дизъюнкции), умножения и инверсии матриц. Реализовать методы для подсчета числа единиц в матрице и упорядочения строк в лексикографическом порядке.

30. Построить класс Булев вектор (BoolVector). Реализовать методы для выполнения поразрядных конъюнкции, дизъюнкции и отрицания векторов, а также подсчета числа единиц и нулей в векторе.

***Задание к лабораторной работе №3***

*Работа со строками. Чтение исходной строки осуществлять из текстового файла, для результирующей строки применить стандартный алгоритм шифрования и записать в файл. При выполнении следующих заданий для вывода результатов создавать новую директорию и файл средствами класса File.*

В каждом слове текста k-ю букву заменить заданным символом. Если k больше длины слова, корректировку не выполнять.

2. В тексте каждую букву заменить ее порядковым номером в алфавите. При выводе в одной строке печатать текст с двумя пробелами между буквами, в следующей строке внизу под каждой буквой печатать ее номер.

3. В тексте после буквы Р, если она не последняя в слове, ошибочно напечатана буква А вместо О. Внести исправления в текст.

4. В тексте после k-го символа вставить заданную подстроку.

5. После каждого слова текста, заканчивающегося заданной подстрокой, вставить указанное слово.

6. В зависимости от признака (0 или 1) в каждой строке текста удалить указанный символ везде, где он встречается, или вставить его после k-гo символа.

7. Из текста удалить все символы, кроме пробелов, не являющиеся буквами. Между последовательностями подряд идущих букв оставить хотя бы один пробел.

8. Удалить из текста его часть, заключенную между двумя символами, которые вводятся (например, между скобками «(» и «)» или между звездочками «\*» и т.п.).

9. Определить, сколько раз повторяется в тексте каждое слово, которое встречается в нем.

10. В тексте найти и напечатать n символов (и их количество), встречающихся наиболее часто. 11. Найти, каких букв, гласных или согласных, больше в каждом предложении текста.

12. В стихотворении найти количество слов, начинающихся и заканчивающихся гласной буквой.

13. Напечатать без повторения слова текста, у которых первая и последняя буквы совпадают. 14. В тексте найти и напечатать все слова максимальной и все слова минимальной длины.

15. Напечатать квитанцию об оплате телеграммы, если стоимость одного слова задана.

16. В стихотворении найти одинаковые буквы, которые встречаются во всех словах.

17. В тексте найти первую подстроку максимальной длины, не содержащую букв.

18. В тексте определить все согласные буквы, встречающиеся не более чем в двух словах.

19. Преобразовать текст так, чтобы каждое слово, не содержащее неалфавитных символов, начиналось с заглавной буквы.

20. Подсчитать количество содержащихся в данном тексте знаков препинания.

21. В заданном тексте найти сумму всех встречающихся цифр.

22. Из кода Java удалить все комментарии (//, /\*, /\*\*).

23. Все слова текста встречаются четное количество раз, за исключением одного. Определить это слово. При сравнении слов регистр не учитывать.

24. Определить сумму всех целых чисел, встречающихся в заданном тексте.

25. Из текста удалить все лишние пробелы, если они разделяют два различных знака препинания, и если рядом с ними находится еще один пробел.

26. Строка состоит из упорядоченных чисел от 0 до 100000, записанных подряд без пробелов. Определить, что будет подстрокой от позиции n до m.

27. Определить количество вхождений заданного слова в текст, игнорируя регистр символов и считая буквы «е», «ё», и «и», «й» одинаковыми.

28. Преобразовать текст так, чтобы только первые буквы каждого предложения были заглавными.

29. Заменить все одинаковые рядом стоящие в тексте символы одним символом.

30. Вывести в заданном тексте все слова, расположив их в алфавитном порядке.

31. Подсчитать, сколько слов в заданном тексте начинается с прописной буквы.

32. Подсчитать, сколько раз заданное слово входит в текст.

***Задание к лабораторной работе №4***

*Разработать проект управления процессами на основе создания и реализации интерфейсов для предметных областей, создать консольное приложение, удовлетворяющее следующим требованиям: использовать возможности ООП: классы, наследование, полиморфизм, инкапсуляция. Аргументировать принадлежность классу каждого создаваемого метода и корректно переопределить для каждого класса методы equals(), hashCode(), toString(). Каждый класс должен иметь отражающее смысл название и информативный состав. Наследование должно применяться только тогда, когда это имеет смысл. При кодировании должны быть использованы соглашения об оформлении кода java code convention. Классы должны быть грамотно разложены по пакетам. При необходимости использовать внутренние и вложенные классы. Консольное меню должно быть минимальным. Для хранения параметров инициализации можно использовать файлы. Для ввода данных использовать обработку исключительных ситуаций.*

1. Работа с текстом. Определить иерархию содержания текста, используя классы Главы, Пункты. Методы: добавить пункт, вывести на консоль содержание, общее количество страниц. Организовать поиск в содержании по ключевым словам.

2. Каталог автомобилей. Определить иерархию каталога автомобилей, используя классы Марка, Модель, Вид авто, год выпуска. Создать список моделей по определенной марке, провести сортировку авто по видам. Вывести по году выпуска все соответствующие автомобили.

3. Работа с файлами. Определить иерархию директории, используя классы раздел, папка, файл. Организовать поиск файлов по определенному расширению, реализовать расчет объема занимаемой памяти файлами в папке, разделе.

4. Каталог комплектующих ПК (ноутбук, нетбук). Определить иерархию комплектующих ПК. Создать по запросу модель ПК. Подсчитать стоимость. Вывести ПК по запрошенному диапазону памяти, видеокарте.

5. Звукозапись. Определить иерархию музыкальных композиций. Записать на диск сборку. Подсчитать продолжительность. Провести перестановку композиций диска на основе принадлежности к стилю. Найти композицию, соответствующую заданному диапазону длины треков.

6. Транспорт. Определить иерархию подвижного состава железнодорожного транспорта. Создать пассажирский поезд. Подсчитать общую численность пассажиров и багажа. Провести сортировку вагонов поезда на основе уровня комфортности. Найти в поезде вагоны, соответствующие заданному диапазону параметров числа пассажиров.

7. Авиакомпания. Определить иерархию самолетов. Создать авиакомпанию. Посчитать общую вместимость и грузоподъемность. Провести сортировку самолетов компании по дальности полета. Найти самолет в компании, соответствующий заданному диапазону параметров потребления горючего.

8. Таксопарк. Определить иерархию легковых автомобилей. Создать таксопарк. Подсчитать стоимость автопарка. Провести сортировку автомобилей парка по расходу топлива. Найти автомобиль в компании, соответствующий заданному диапазону параметров скорости.

9. Страхование. Определить иерархию страховых обязательств. Собрать из обязательств перечень. Подсчитать стоимость. Провести сортировку обязательств в перечне на основе уменьшения степени риска. Найти обязательство в перечне, соответствующее заданному диапазону параметров.

10. Мобильная связь. Определить иерархию тарифов мобильной компании. Создать список тарифов компании. Подсчитать общую численность клиентов. Провести сортировку тарифов на основе размера абонентской платы. Найти тариф в компании, соответствующий заданному диапазону параметров.

11. Фургон кофе. Загрузить фургон определенного объема грузом на определенную сумму из различных сортов кофе, находящихся к тому же в разных физических состояниях (зерно, молотый, растворимый в банках и пакетиках). Учитывать объем кофе вместе с упаковкой. Провести сортировку товаров на основе соотношения цены и веса. Найти в фургоне товар, соответствующий заданному диапазону параметров качества.

12. Налоги. Определить множество и сумму налоговых выплат физического лица за год с учетом доходов с основного и дополнительного мест работы, авторских вознаграждений, продажи имущества, получения в подарок денежных сумм и имущества, переводов из-за границы, льгот на детей и материальной помощи. Провести сортировку налогов по сумме.

13. Счета. Клиент может иметь несколько счетов в банке. Учитывать возможность блокировки/разблокировки счета. Реализовать поиск и сортировку счетов. Вычисление общей суммы по счетам. Вычисление суммы по всем счетам, имеющим положительный и отрицательный балансы отдельно.

14. Туристические путевки. Сформировать набор предложений клиенту по выбору туристической путевки различного типа (отдых, экскурсии, лечение, шопинг, круиз и т.д.) для оптимального выбора. Учитывать возможность выбора транспорта, питания и числа дней. Реализовать выбор и сортировку путевок.

15. Кредиты. Сформировать набор предложений клиенту по целевым кредитам различных банков для оптимального выбора. Учитывать возможность досрочного погашения кредита и/или увеличения кредитной линии. Реализовать выбор и поиск кредита.

16. Тренажерный зал. Сформировать набор предложений клиенту по выбору вида занятия (степ, фитнес, силовая нагрузка, индивидуальное занятие и т.д.) для оптимального выбора. Учитывать возможность выбора дней недели, времени и стоимости. Реализовать выбор и сортировку занятий, сделать расчет стоимости абонемента с учетом скидки.

17. Банковские карты. Клиент может иметь несколько банковских карт в одном банке. К одному счету может привязано несколько карт, карты могут быть в разных валютах. Реализовать поиск и сортировку карт по сроку. Вычисление общей суммы по счету в разных валютах. Вычисление суммы по всем счетам, имеющим положительный и отрицательный балансы отдельно.

18. Каталог мобильных телефонов. Определить иерархию каталога телефонов, используя классы Производитель, Модель, Вид телефона, год выпуска. Создать список телефонов по определенному производителю, провести сортировку по заданным параметрам. Вывести по году выпуска все соответствующие мобильные телефоны.

19. Разработка IT – проекта. Основные фазы разработки проекта: разработка ТЗ, проектирование, кодирование, тестирование, пробная эксплуатация. Создать иерархию проекта, назначить на каждую фазу группу, в которой будет по 2 человека. Создать график сдачи работ по проекту согласно последовательности разработки. Реализовать поиск информации по заданным параметрам. (ФИО, фаза и т.д.).

20. Каталог мебели. Определить иерархию каталога мебели, используя классы: Производитель, Назначение, Вид мебели, цена. Создать список продукции по определенному производителю, провести сортировку по заданным параметрам. Создать список по назначению с сортировкой по цене.

21.Каталог ПО (папка на диске). Определить иерархию директории, используя классы: вид ПО, название, версия. Организовать поиск файлов по определенному виду, реализовать расчет объема занимаемой памяти файлами в папке, вывести иерархию версий по названию ПО.

22. Система оплаты. Выбор способа оплаты (Paypal, банковская карта (Visa, Mastercard), Ерип (QR код). Каждому способу оплаты соответствует свой набор атрибутов. Создать личный кабинет пользователя с привязкой банковских карт и видов оплаты, организовать поиск оплат по способу оплаты.

23. Социальная сеть. Личные страницы пользователей(Id), перечень сервисов, предоставляемых пользователю. Организовать чат между пользователями, осуществить поиск сообщений по ключевым словам.

24. Логистический склад. Определенному товару соответствуют сектор, ряд, место, дата поступления, количество. Вывести по заданным параметрам (сектор, ряд, место) соответствующую информацию о товаре. Создать список товаров, поступивших за определенный период времени.

25. Курьерская доставка. Каждому курьеру соответствует список заказов на день для доставки по времени. Информация по заказу: номер, состав, стоимость, статус, дата. Вывести заказы, которые доставлены на текущее время, не скомплектованы, не доставлены вовремя. Рассчитать зарплату курьера за день, исходя из определенных тарифов за доставку(определить).

26. Электронная зачетная книжка. Определить средний балл студента по сессиям. Осуществить поиск по заданным параметрам (по фамилии преподавателя, по предмету и т.д.). Основные предполагаемые классы: дисциплины, семестр, сессия.

27. Деканат. Вывести средний балл за сессию по соответствующей группе. Основные предполагаемые классы: специальность, дисциплины, группа. Сформировать список студентов по факультету, не сдавших сессию.

28. Лента новостей. Ориентировочные классы: категория (главная, последняя, срочная), рубрика (политика, финансы и т.д.), новость (название, автор, время т.д.). Вывести список новостей по

времени поступления. Организовать поиск по параметрам.

29. Конференция. Ориентировочные классы: секция (название, комиссия, перечень докладов, время, место и т.д.), автор, доклад. Вывести расписание докладов по секциям. Организовать поиск по параметрам.

30. Локальная сеть. Ориентировочные классы: ПК (имя, IP, и т.д.), сервер (имя, IP, и т.д.), домен (имя, расположение и т.д.). Вывести список ПК определенного домена, организовать поиск по заданным параметрам.

***Задание к лабораторной работе №5***

*Используя Collection Framework (соответствующий интерфейс и Stream API):*

1. Ввести строки из файла, записать в список. Вывести строки в файл в обратном порядке.

2. Ввести число, занести его цифры в стек. Вывести число, у которого цифры идут в обратном порядке.

3. Создать список из элементов каталога и его подкаталогов.

4. Занести статьи одного автора в список. Провести сортировку по возрастанию длин строк.

5. Задать два стека, поменять информацию местами.

6. Определить множество на основе множества целых чисел. Создать методы для определения пересечения и объединения множеств.

7. Списки, стеки или очереди T(1..n) и U(1..n) содержат результаты n-измерений тока и напряжения на неизвестном сопротивлении R. Найти приближенное число R методом наименьших квадратов.

8. С использованием множества выполнить попарное суммирование произвольного конечного ряда чисел по следующим правилам: на первом этапе суммируются попарно рядом стоящие числа, на втором этапе суммируются результаты первого этапа и т.д. до тех пор, пока не останется одно число.

9. Сложить два многочлена заданной степени, если коэффициенты многочленов хранятся в объекте HashMap.

10. Умножить два многочлена заданной степени, если коэффициенты многочленов хранятся в различных списках.

11. Не используя вспомогательных объектов, переставить отрицательные элементы данного списка в конец, а положительные — в начало списка.

12. Ввести строки из файла, записать в список ArrayList. Выполнить сортировку строк, используя метод sort() из класса Collections.

13. Задана строка, состоящая из символов «(», «)», «[», «]», «{», «}». Проверить правильность расстановки скобок. Использовать стек.

14. Задан файл с текстом на английском языке. Выделить все различные слова. Слова, отличающиеся только регистром букв, считать одинаковыми. Использовать класс HashSet.

15. Задан файл с текстом на английском языке. Выделить все различные слова. Для каждого слова подсчитать частоту его встречаемости. Слова, отличающиеся регистром букв, считать различными. Использовать класс HashMap.

16. Заполнить HashMap 10 объектами. Найти строки, у которых ключ>5. Если ключ = 0, вывести строки через запятую. Перемножить все ключи, где длина строки>5.

17. Написать функцию, которая получала бы итераторы на начало и конец отсортированного List и заданный символ. Возвращать функция должна начало и конец диапазона, строки в котором начинаются с заданного символа.

18. Задан список целых чисел и некоторое число X. Не используя вспомогательных объектов и методов сортировки и не изменяя размера списка, переставить элементы списка так, чтобы сначала шли числа, не превосходящие X, а затем числа, больше X.

19. Написать программу, осуществляющую сжатие английского текста. Построить для каждого слова в тексте оптимальный префиксный код по алгоритму Хаффмена. Использовать класс PriorityQueue.

20. Реализовать класс Graph, представляющий собой неориентированный граф. В конструкторе класса передается количество вершин в графе. Методы должны поддерживать быстрое добавление и удаление ребер.

21. На базе коллекций реализовать структуру хранения чисел с поддержкой следующих операций:

• добавление/удаление числа;

• поиск числа, наиболее близкого к заданному (т.е. модуль разницы минимален).

22. Во входном файле расположены два набора положительных чисел; между наборами стоит отрицательное число. Построить два списка C1 и С2, элементы которых содержат соответственно числа 1-го и 2-го набора таким образом, чтобы внутри одного списка числа были упорядочены по возрастанию. Затем объединить списки C1 и С2 в один упорядоченный список, изменяя только значения полей ссылочного типа.

23. На плоскости задано N точек. Вывести в файл описания всех прямых, которые проходят более чем через одну точку из заданных. Для каждой прямой указать, через сколько точек она проходит. Использовать класс HashMap.

24. На плоскости задано N отрезков. Найти точку пересечения двух отрезков, имеющую минимальную абсциссу. Использовать класс TreeMap.

25. Дана матрица из целых чисел. Найти в ней прямоугольную подматрицу, состоящую из максимального количества одинаковых элементов. Использовать класс ArrayDeque.

26. Во входном файле хранятся наименования некоторых объектов. Построить список Z, элементы которого содержат наименования и шифры данных объектов, причем элементы списка должны быть упорядочены по возрастанию шифров. Затем «сжать» список Z, удаляя дублирующие наименования объектов.

27. Не используя вспомогательных объектов, переставить положительные элементы данного списка в конец, а отрицательные — в начало списка.

28. На плоскости задано N точек. Вывести в файл описания всех прямых, которые проходят только через одну точку из заданных. Для каждой прямой указать, через какую точку она проходит. Использовать класс HashMap.

29. Ввести строки из файла, записать в список. Вывести строки в файл в алфавитном порядке (по первой букве строки).

30. На плоскости задано N отрезков. Найти точку пересечения двух отрезков, имеющую минимальную ординату. Использовать класс TreeMap.

***Задание к лабораторной работе №6***

*Разработать многопоточное приложение. Использовать возможности, предоставляемые пакетом java.util.concurrent. Не использовать слово synchronized. Все сущности, желающие получить доступ к ресурсу, должны быть потоками.*

1. Порт. Корабли заходят в порт для разгрузки/загрузки контейнеров. Число контейнеров, находящихся в текущий момент в порту и на корабле, должно быть неотрицательным и превышающим заданную грузоподъемность судна и вместимость порта. В порту работает несколько причалов. У одного причала может стоять один корабль. Корабль может загружаться у причала, разгружаться или выполнять оба действия.

2. Маленькая библиотека. Доступны для чтения несколько книг. Одинаковых книг в библиотеке нет. Некоторые выдаются на руки, некоторые только в читальный зал. Читатель может брать на руки и в читальный зал несколько книг.

3. Автостоянка. Доступно несколько машиномест. На одном месте может находиться только один автомобиль. Если все места заняты, то автомобиль не станет ждать больше определенного времени и уедет на другую стоянку.

4. CallCenter. В организации работает несколько операторов. Оператор может обслуживать только одного клиента, остальные должны ждать своей очереди. Клиент может положить трубку и перезвонить еще раз через некоторое время.

5. Автобусные остановки. На маршруте несколько остановок. На одной остановке может останавливаться несколько автобусов одновременно, но не более заданного числа.

6. Свободная касса. В ресторане быстрого обслуживания есть несколько касс. Посетители стоят в очереди в конкретную кассу, но могут перейти в другую очередь при уменьшении или исчезновении там очереди.

7. Тоннель. В горах существует два железнодорожных тоннеля, по которым поезда могут двигаться в обоих направлениях. По обоим концам тоннеля собралось много поездов. Обеспечить безопасное прохождение тоннелей в обоих направлениях. Поезд можно перенаправить из одного тоннеля в другой при превышении заданного времени ожидания на проезд.

8. Банк. Имеется банк с кассирами, клиентами и их счетами. Клиент может снимать/пополнять/переводить/оплачивать/обменивать денежные средства. Кассир последовательно обслуживает клиентов. Поток-наблюдатель следит, чтобы в кассах всегда были наличные, при скоплении денег более определенной суммы, часть их переводится в хранилище, при истощении запасов наличных происходит пополнение из хранилища.

9. Аукцион. На торги выставляется несколько лотов. Участники аукциона делают заявки. Заявку можно корректировать в сторону увеличения несколько раз за торги одного лота. Аукцион определяет победителя и переходит к следующему лоту. Участник, не заплативший за лот в заданный промежуток времени, отстраняется на несколько лотов от торгов.

10. Биржа. На торгах брокеры предлагают акции нескольких фирм. На бирже совершаются действия по купле-продаже акций. В зависимости от количества проданных-купленных акций их цена изменяется. Брокеры предлагают к продаже некоторую часть акций. От активности и роста-падения котировок акций изменяется индекс биржи. Биржа может приостановить торги при резком падении индекса.

11. Аэропорт. Посадка/высадка пассажиров может осуществляться через конечное число терминалов и наземным способом через конечное число трапов. Самолеты бывают разной вместимости и дальности полета. Организовать функционирование аэропорта, если пунктов назначения 4–6, и зон дальности 2–3.

12. Гипермаркет. В кассовой зоне есть несколько касс самообслуживания. Покупатели стоят в общей очереди к этим кассам, при освобождении кассы предыдущим покупателем, она занимается покупателем, следующим по очереди.

13. Электронная очередь в отделении почты. Есть несколько окон обслуживания, услуги имеют категорию. 1 окно обслуживает только 1 и 2 категорию, 2 окно обслуживает 2 и 3 категорию, остальные окна обслуживают все категории. Организовать работу почтового отделения.

14. Визовый центр. Имеется поток посетителей на подачу виз разных категорий. На входе консультант проверяет наличие необходимого пакета документов, далее поток распараллеливается в зависимости от вида визы к определенному консультанту, после оформления у консультанта посетители производят оплату в 2-х кассовых окнах. Организовать работу визового центра.

15. Гибкий производственный модуль. На конвейере детали поступают к 3 многоканальным прессам, время прессования у которых разное, далее отпрессованные детали поступают на сборку к производственному модулю по одной от каждого пресса, из них собирается полноценная деталь. Определить производительность ГПМ за смену.

16. Техническая поддержка абонентов. Абоненты IPTV обращаются к специалистам техподдержки 2 способами: по телефону и онлайн – чат. В зависимости от проблемы клиенты делятся на 3 категории, каждый специалист может одновременно обслуживать только 1 клиента. Организовать работу технической поддержки.

17. Автоматизированный склад. Машины занимают место для разгрузки/загрузки. Имеется три места разгрузки/загрузки. На одном месте разгружается один авто. Авто может загружаться на складе, разгружаться или выполнять оба действия. Организовать работу склада.

18. Автоматизированная линия. Детали поступают к 3 станкам и распределяются к тем станкам, которые на данный момент свободны, или у которых очередь минимальна. Определить производительность каждого станка за смену.

19. Spa - салон. Клиентам доступен перечень процедур. Клиент может выбрать несколько услуг. Представлен прайслист. Некоторые процедуры относятся к косметологическим, некоторые к медицинским. Клиент может выбрать несколько процедур одной и второй категории. Организовать работу спа – салона, вывести чек по заказанным услугам.

20. Стоматологическая клиника. Представлены следующие классификации врачей: терапевт, ортопед, хирург. Пациент в один день может посетить только 2-х врачей, у каждого врача свой перечень услуг. Организовать работу клиники, вывести список услуг и посещений врача одним клиентом за неделю.

21. Сервис покупки билетов. Имеется перечень рейсов на определенное направление. Пользователи совершают бронирование. Бронирование можно корректировать в сторону увеличения и уменьшения количества билетов. Бронирование действует 10 минут. Если не поступает оплата в течение этого времени, другой пользователь может забронировать эти же билеты.

22. Кассовое обслуживание банка. В кассовой зоне банка имеется несколько касс. В одной кассе обслуживается несколько видов операций, но транзакции должны иметь каждая свою очередь для фиксации и записи в системе. Организовать работу касс банка.

23. Нотариальные услуги. В нотариальной конторе ведут прием несколько нотариусов (каждый обслуживает 2 категории). Для посетителей организована электронная очередь согласно категории услуги (всего 6). Когда нотариус заканчивает работу с предыдущим клиентом, следующий клиент перенаправляется к нему. Организовать работу нотариальной конторы.

24. Страховая компания. Работают 3 менеджера, имеется поток клиентов. Каждый менеджер работает по двум видам услуг (всего 5). Клиент может быть обслужен разными менеджерами согласно услуге. Организовать работу страховой компании.

25. Автоматизированная мойка авто. Имеется 2 портала с набором видов мойки авто. Каждому виду мойки соответствует определенное время. Имеется общая очередь к порталам, клиент по очереди въезжает в тот портал, который на данный момент освобождается. Организовать работу мойки.

26. Таможенный терминал. Общая очередь в терминале делится на 2 красных и один зеленый канал прохода авто, среднее время прохождения на зеленом канале в 2 раза меньше времени прохождения на красном. Организовать работу терминала.

27. Медицинский центр. Имеется перечень услуг. Каждый врач выполняет прием согласно одной услуге. Имеется поток посетителей, один пациент может заказать необходимый перечень услуг. Организовать работу центра.

28. Аптека. Доступен перечень лекарств. Одинаковых наименований нет. Определенные наименования лекарств реализуются по рецепту, остальные – в свободной продаже. Покупатель может купить несколько наименований как по рецепту, так и лекарства без рецепта. Сформировать чек покупателю.

29. Тестирование производительности. К определенному сайту подключается несколько пользователей (max=5) последовательно через каждый 10 с, сессия работы у каждого (15+номер пользователя) секунд, реализовать работу теста.

30. Техническая поддержка оператора связи. Имеется поток звонков от клиентов. На первом этапе работает робот-автоответчик, который перенаправляет клиентов согласно проблеме, либо завершает звонок. На звонки отвечают 3 специалиста. Оператор может обслуживать только одного клиента, остальные должны ждать своей очереди. Клиент может положить трубку и перезвонить еще раз через некоторое время.

***Задание к лабораторной работе №7***

*Использовать предметную область по варианту из задания к лабораторной работе №4. В каждом из заданий необходимо выполнить следующие действия:*

*• организацию соединения с базой данных вынести в отдельный класс, метод которого возвращает соединение;*

*• создать БД. Привести таблицы к одной из нормальных форм;*

*• создать класс для выполнения запросов на извлечение информации из БД с использованием компилированных запросов;*

*• создать класс на модификацию информации.*

*Создать на основе сокетов клиент/серверное приложение:*

1. Клиент посылает через сервер сообщение другому клиенту, выбранному из списка.

2. Чат. Клиент посылает через сервер сообщение, которое получают все клиенты. Список клиентов хранится на сервере в файле.

3. Клиент при обращении к серверу получает случайно выбранное задание из файла.

4. Сервер рассылает сообщения выбранным из списка клиентам. Список клиентов хранится в файле.

5. Сервер рассылает сообщения в определенное время определенным клиентам.

6. Сервер рассылает сообщения только тем клиентам, которые в настоящий момент находятся в on-line.

7. Чат. Сервер рассылает всем клиентам информацию о клиентах, вошедших в чат и покинувших его.

8. Клиент выбирает изображение из списка и пересылает его другому клиенту через сервер. Написать программу, сканирующую сеть в указанном диапазоне IP-адресов на наличие активных компьютеров.

9. Прокси. Программа должна принимать сообщения от любого клиента, работающего на протоколе TCP, и отсылать их на соответствующий сервер. При передаче изменять сообщение.

10. Создать консольное приложение-калькулятор, используя вычислительную среду Google Sheets.

11. Написать программу, сканирующую сеть в указанном диапазоне IP-адресов на наличие активных компьютеров.

12. Прокси. Программа должна принимать сообщения от любого клиента, работающего на протоколе TCP, и отсылать их на соответствующий сервер. При передаче изменять сообщение.