Список задач

**(все приложения собирать с помощью Apache Ant (или Maven). Для задач должны быть тесты, для тестирования использовать jUnit. Соблюдение Java Code Convention обязательно)**

1. Создать простой ssh клиент в виде апплета.
2. Создать ftp сервер использовать rfc 959. Использовать сокеты.
3. Создать графический редактор по типу paint (имеющего клиент –серверную архитектуру) в виде апплета, при этом настройки графического интерфейса передаются в виде XML файла. Предусмотреть возможность сохранения и восстановления созданного.
4. Создать web страницу , которая отображается по разному для разных браузеров, при этом контент передается в виде XML файла, иерархия задается в DTD (xsd) файле, контент отображается с использованием xsl файла, который генерируется динамически.
5. Используя сервлеты и БД Oracle, создать Web приложение, которое запрашивает у клиента имя и пароль (должны передаваться на сервер в шифрованном виде), затем отображает список файлов, которые клиент может загрузить, при этом каждому зарегистрированному клиенту могут отображаться различные списки файлов. Параметры авторизации хранить в сессии. Для работы с БД использовать JDBC.
6. Создать приложение, которое может отправлять почту по указанному адресу, а также ее принимать. Приложение должно иметь графический интерфейс.(использовать javax.mail)
7. Используя WebGL реализовать 3D диаграмму Вороного для набора точек. Точки вводятся посредством Web формы. Логика выполняется на сервере, данные клиенту передаются с использованием WebSockets.
8. Создать компонент JavaBeans для NetBeans 7.0 или Eclipse. Визуальный компонент для работы с базой данных. Должно отображаться содержимое соответствующей таблицы базы данных, а также должна быть возможность изменения данных в таблице. Обязательно предусмотреть редактор свойств.
9. Создать классы пользовательских тегов, формирующих нужное количество элементов (строк, ячеек и др.) для размещения результатов выполнения запроса. Элемент массива называют локальным максимумом, если у него нет соседа большего, чем он сам. Аналогично определяется локальный минимум. Определить количество локальных максимумов и локальных минимумов в заданном строкой массиве чисел. Массив задает клиент. Возвратить все максимумы и минимумы пользователю.

1. Используя механизм сериализации, создать клиент серверное приложение: Клиент создает объект выполняет его сериализацию пересылает на сервер, сервер производит десериализацию и загружает его в память.
2. Используя потоки, решить систему линейных уравнений с трехдиагональной матрицей.
3. Напишите метод, принимающий в качестве параметра ссылку на объект группы потоков и создающий поток, который периодически выводит на экран данные об иерархии потоков и подгрупп потоков внутри заданной группы. Протестируйте метод в составе программы, использующей несколько непродолжительных по времени выполнения потоков, которые принадлежат различным группам.
4. Используя механизм рефлексии, напишите программу, которая выводит на экран полное описание класса с заданным именем, включающее всю информацию о классе за исключением инструкций импорта, комментариев и кода инициализаторов, конструкторов и методов. Для загрузки класса использовать собственный загрузчик.
5. Создать апплет летающий шарик, который отражается от стенок, а также летающий за курсором мыши.
6. Создайте пару потоковых классов Filter*,* EncryptOutputStreamи DecryptlnputStream**,** выполняющих шифрование и дешифрование исходной последовательности байтов в соответствии с любым алгоритмом по вашему выбору (например, предусматривающим выполнение операции XOR над каждым отдельным байтом с использованием некоторого фиксированного значения), чтобы объект класса DecryptlnputStreamбыл способен дешифровать байты, зашифрованные потоком типа EncryptOutputStream**.**
7. Изобразить в приложении правильные треугольники, вращающиеся в плоскости экрана вокруг своего центра. Каждому объекту соответствует поток с заданным приоритетом. Через каждые 5 сек. Производить синхронизацию вращения с использованием системы уведомлений JavaBeans.
8. Используя авторизованный вход в приложение(параметры авторизации, дату входа в приложение и время работы сохранять в сессии) Подсчитать количество различных слов в файлах с суффиксом, который задается с клавиатуры и сохранить информацию в файл cookie. Для поиска соответствующих файлов использовать средства интерфейсов FilenameFilter и FileFilter.
9. Создать сервлет и взаимодействующие с ним пакеты Java-классов и  
   HTML-документов, выполняющие следующие действия:  
   Осуществить форматирование выбранного пользователем текстового файла, так   
   чтобы все абзацы имели отступ ровно 3 пробела, а длина каждой строки была   
   ровно 80 символов и не имела начальными и конечными символами пробел.
10. Создать классы пользовательских тегов, формирующих нужное количество элементов (строк, ячеек и др.) для размещения результатов выполнения запроса. В базе данных хранятся координаты конечного множества точек плоскости. Пользователем вводятся координаты центра и радиусы 5 концентрических окружностей. Между какими окружностями (1 и 2, 2 и 3, ..., 4 и 5) больше всего точек заданного множества? Полученное множество точек возвратить клиенту.
11. Написать на Java IRC клиент (использовать rfc1459)
12. Написать сценарий, в котором при заходе на jsp (использовать jstl) страницу, клиент вводит в поле формы путь к файлу и отсылает ее на сервер. Если путь корректен, то сервер передает соответствующий файл кленту
13. Написать GUI приложение, которое читает по http указанные файлы (возможно неограниченно). Предусмотреть возможность запуска и останова процесса. Результаты записываются в БД (использовать MySQL), а затем формирует отчет в виде документа Excel:   
      
    Имя файла успешность прочтения ошибка   
    Для создания отчета использовать JasperReports.
14. Изобразить точку, пересекающую с постоянной скоростью окно слева направо (справа налево) параллельно горизонтальной оси. Как только точка доходит до границы окна, в этот момент от левого (правого) края с вертикальной координатной *y*, выбранной с помощью датчика случайных чисел, начинает свое движение другая точка и т.д. Цвет точки также можно выбирать с помощью

датчика случайных чисел. Для каждой точки создается собственный поток.

1. Создать форму с выпадающим списком так, чтобы при выборе элемента списка на экране появлялись GIF-изображения, двигающиеся в случайно выбранном направлении по апплету.
2. Создать Web чат с использованием технологий Servlets или JSP и БД Oracle При этом для идентификации клиентов использовать Cookie Приложение развернуть на сервере приложений IBM WebSphere или BEA WebLogic или Tomcat. В качестве СУБД использовать Oracle. Клиентская часть должна быть реализована с помощью GWT. Для работы с базой данных использовать Hibernate.
3. Создать компонент JavaBeans для NetBeans 7.0 или Eclipse. Визуальный компонент медиаплеер, который проигрывает avi файлы. Звук не обязателен, но желателен. Должна быть предусмотрена привязка проигрываемого файла в свойствах компонента.