1. 跳表的实现:
数据结构的设计，基本操作查找、插入、删除的实现

2. K叉哈夫曼树的实现，完成编码及压缩。

3. 红黑树（伸展树、TRIE树等）的建立、查找、删除算法。

4. 散列提高部分：hash一致性问题、安全支付问题

5. 线性排序算法的探讨，主要设计性能测试比较程序:实验设计、性能曲线（一般情况，正序，逆序，大数据，一般数据等）

hash一致性问题

问题描述： 例如手机朋友网有n个服务器，为了方便用户的访问会在服务器上缓存数据，因此用户每次访问的时候最好能保持同一台服务器。已有的做法是根据ServerIPIndex[QQNUM%n]得到请求的服务器，这种方法很方便将用户分到不同的服务器上去。但是如果一台服务器死掉了，那么n就变为了n-1，那么ServerIPIndex[QQNUM%n]与ServerIPIndex[QQNUM%（n-1）]基本上都不一样了，所以大多数用户的请求都会转到其他服务器，这样会发生大量访问错误。

问： 如何改进或者换一种方法，使得：

（1）一台服务器死掉后，不会造成大面积的访问错误，

（2）原有的访问基本还是停留在同一台服务器上；

（3）尽量考虑负载均衡。

6. B-Trees的实现及分析

基本要求

- ①　实现在B-树上的查找，并分析其时间复杂性。
- ②　实现B-树的ADT，包括其上的基本操作：结点的加入和删除。
- ③　要求B-树结构中的M=3或5，实现其中的一种即可。
- ④　实现基本操作的演示。

实现提示

- 主要考虑结点的分裂和和并。

7. Binomial heaps的实现和分析

- 基本要求
- ① 设计二项堆ADT，其上的基本操作包括：Make Heap (x), Find-Min, Union，Insert，Extract-Min，Decrease Key (x)，Delete。
- ② 实现二项堆ADT，包括实现二项堆的存储结构以及其上的基本操作，并分析基本操作的时间复杂性。

8. 蚁群算法在旅行商问题中的应用

- 基本要求
- ①　应用蚁群算法求解TSP问题。
- ②　TSP中的城市数量不少于30个，组成完全图，边上的权值自定。
- ③　蚂蚁数量可配置，迭代次数可配置。
- ④　给出较全面的实验结果：结果路经及长度；蚁群算法执行时间；不同参数值（蚂蚁数量，迭代次数）的影响等。

9.长整数的代数计算

- (1) 问题描述
- 设计数据结构完成长整数的表示和存储，并编写算法来实现两长整数的加、减、乘、除等基本代数运算。
- (2) 课程设计目的
- 能够应用线性数据结构解决实际问题。
- (3) 基本要求
- ①　　长整数长度在一百位以上。
- ②　　实现两长整数在取余操作下的加、减、乘、除操作，即实现算法来求解a+b mod n, a-b mod n, a　b mod n, a　b mod n。
- ③　　输入输出均在文件中。
- ④　　分析算法的时空复杂性。
- (4) 实现提示
- 需将长整数的加法转化为多个一般整数加法的组合。

10. 实现并对比三种基本的字符串匹配算法

- 基本要求
- ① 编程实现三种字符串匹配算法。
- ② 分析三种算法的时间复杂性，通过应用三种算法在一个较长英文文本中查找一个子串的实验数据来对比三种算法的运行时间。

11. 应用堆实现一个优先队列并实现作业的优先调度

- 基本要求
- ① 给出优先队列的ADT描述，包括队列的逻辑结构及其上基本操作。
- ② 以堆结构为辅助结构实现优先队列的存储表示并实现其上的基本操作。
- ③ 作业集合中的各作业随机生成，根据作业的s属性和e属性动态调整作业队列，不断加入作业，作业结束删除作业。
- ④ 要对作业调度的结果给出清晰的输出信息，包括：何和作业进入，何时调度哪个作业，何时离开，每个作业等待多长时间，优先数的动态变化情况等。

- 应用该优先队列实现作业的优先调度：
- 一个作业ti =（si，ei），si为作业的开始时间（进入时间），ei为作业的结束时间（离开时间）。作业调度的基本任务是从当前在系统中的作业中选取一个来执行，如果没有作业则执行nop操作。本题目要求的作业调度是基于优先级的调度，每次选取优先级最高的作业来调度，优先级用优先数（每个作业一个优先数pi）表征，优先数越小，优先级越高。作业ti进入系统时，即si时刻，系统给该作业指定其初始优先数pi = ei - si，从而使越短的作业优先级越高。该优先数在作业等待调度执行的过程中会不断减小，调整公式为：pi = pi - wi，其中的wi为作业ti的等待时间：wi = 当前时间-si。一旦作业被调度，该作业就一直执行，不能被抢占，只有当前执行作业指向完成时，才产生下一轮调度。所以可以在每次调度前动态调整各作业的优先数。
- 编程实现这样一个作业调度系统。