**设计思路：**

假设有两个操作对象分别为R(X,Y)和S(Y,Z)，对R和S做散列函数计算，分别得到散列划分后，用块嵌套循环的方式分别连接相应划分。

**实现细节：**

确定连接属性Y上的散列函数h，h尽可能平均和随机；将2(n+1)个桶置空，准备用于存放散列划分后的结果；利用h分别计算R和S中各元组连接属性Y的值，得到值i(0<=i<=n)，根据值i将元组放入对应的桶i中，若发生溢出则建立溢出桶，桶与溢出桶之间建立指针连接。对于桶中的内容（包括溢出桶），假设内存缓冲区足够将两个对应桶中较小者的所有块放入，则直接进行块嵌套循环连接：依次取出第二个桶中的元组，根据这个元组连接属性上的值通过提前建立好的内存查找结构进行查询，若能找出满足互相连接条件（对应桶中连接属性相同）的元组，则把连接结果加入到结果表中，不断重复这样的步骤，直到遍历第二个桶中所有的元组为止；若内存缓冲区不足够将两个对应桶中较小者的所有块放入，则通过递归函数对桶进行子桶划分，直到子桶足够小能放入内存中，再进行上述过程。将所有对应桶进行上述操作，最后将连接结果表存入磁盘中。

**数据结构：**

每个元组用结构体tuple储存，里面包括元组的各属性；

用列表存储R和S中的元组，按列表数组顺序逐个进行散列划分；

散列函数得到的划分用指针连接，有一个头指针指向第一个桶，每个桶用结构体bucket储存，bucket中包含上述的tuple数组（用于储存每一个元组）、指向溢出桶的指针added（无溢出桶时为NULL）和指向下一个桶的指针next（最后一个用的next为NULL）；

溢出桶的储存结构与桶大致相同，没有指向下一个桶的指针next；

划分子桶时用一个递归函数进行划分，直到子桶满足条件；

内存查找结构利用AVL树来建立，AVL树结构与实验4中相同。

**文件结构：**

散列文件和顺序文件

散列文件：通过Hash函数转换的键值得到元组的物理地址。

顺序文件：文件中的所有记录按关键字顺序排列。