

第 19 讲模拟练习题解析

1901、已知 B_X 为关系 X 的磁盘块数目， I_X 为关系 X 的元组的长度， T_X 为关系 X 的元组的数目， b 为磁盘块或内存页的大小， M 为可用内存页的数目。问：以下算法的适用条件是_____。

```
For i = 1 to  $B_R$ 
  read i-th block of R ;
Next i
For j = 1 to  $B_S$ 
  read j-th block of S ;
  For p = 1 to  $T_R$ 
    read p-th record of R ;
    For q = 1 to  $b/I_S$ 
      read q-th record of S ;
      if  $R.A \theta S.B$  then
        { 串接R的p-th记录和S的q-th记录
          存入结果关系 ; }
    Next q
  Next p
Next j
```

正确答案：C。解析：如上算法是一次性将 R 完整的读入到内存，将 S 一块一块的读入内存进行处理，因此只要 $B_R < M$ ，则无论 S 有多大都能被处理，因此“ $B_S > B_R$ ， $B_R < M$ ”是正确的。

1902、已知 B_X 为关系 X 的磁盘块数目， I_X 为关系 X 的元组的长度， T_X 为关系 X 的元组的数目， b 为磁盘块或内存页的大小， M 为可用内存页的数目， M_X 为分配给 X 的内存区域。问：不考虑结果关系的存取，以下算法的复杂性是_____。

```

For i = 1 to  $B_S/(M-2)$ 
  read i-th Sub-set of S into  $M_S$  ;
  For j = 1 to  $B_R$ 
    read j-th block of R into  $M_R$  ;
    For p = 1 to  $(M-2)b/I_S$ 
      read p-th record of S ;
      For q = 1 to  $b/I_R$ 
        read q-th record of R ;
        if  $R.A \theta S.B$  then
          { 串接S的p-th记录和R的q-th记录 ;
            存入结果关系 ; }
      Next q
    Next p
  Next j
Next i

```

正确答案：C。解析：该算法一次性读入 S 的 $M-2$ 个存储块，循环处理一遍 R 的每一个存储块。R 被循环读取了 $B_S/(M-2)$ 次，S 被读取了一次。故正确答案为 “ $B_R B_S / (M-2) + B_S$ ”。

1903、已知 B_X 为关系 X 的磁盘块数目， I_X 为关系 X 的元组的长度， T_X 为关系 X 的元组的数目，b 为磁盘块或内存页的大小，M 为可用内存页的数目， M_X 为分配给 X 的内存区域。问：关于以下两个算法(a)(b)，说法正确的是_____。

```

For i = 1 to  $B_R$ 
  read i-th block of R ;
Next i
For j = 1 to  $B_S$ 
  read j-th block of S ;
Next j
For p = 1 to  $T_R$ 
  read p-th record of R ;
  For q = 1 to  $T_S$ 
    read q-th record of S ;
    if  $R.A \theta S.B$  then
      {串接R的p-th记录和S的q-th记录 ;
        存入结果关系 ; }
  Next q
Next p

```

算法(a)

```

For i = 1 to  $B_R$ 
  read i-th block of R ;
Next i
For j = 1 to  $B_S$ 
  read j-th block of S ;
  For p = 1 to  $T_R$ 
    read p-th record of R ;
    For q = 1 to  $b/I_S$ 
      read q-th record of S ;
      if  $R.A \theta S.B$  then
        { 串接 R的p-th记录和S的q-th记录 ;
          存入结果关系 ; }
    Next q
  Next p
Next j

```

算法(b)

正确答案：B。解析：算法(a)和(b)的复杂性都是 $B_R + B_S$ ，是相同的。但适用条件是不同的，算法(a)适用于内存充分大，足以装得下两个关系时才可应用；而算法(b)适用于内存只要能够完整的装下一个关系时即可应用。

1904、下列关系代数操作，任何时候都能够用一趟算法实现的是_____。

正确答案：B。解析：选择操作和包上的并操作，在任何时候都可以用一趟算法实现之。而集合上的并操作则不一定能够用一趟算法实现之。故选项(B)是正确的。

1905、关于给出的九个关系代数操作：

$\cup_S, \cap_S, -_S, \cup_B, \cap_B, -_B, \delta(R), \gamma(R), \tau(R)$,

问任何时候都能够用一趟算法实现的操作的个数是_____。

正确答案：B。解析：这九个操作只有包上的并操作，在任何时候都可以用一趟算法实现之。其他均都不一定能够用一趟算法实现之。故选项(B)是正确的。

1906、已知 R 和 S 的参数 $B_R=B_S=10,000$ ，可用内存页数目为 M。若要进行 R 和 S 的 Theta-连接操作，没有索引可以使用，忽略结果存取的 I/O 次数。问：M 分别应是怎样的值，才能使磁盘的 I/O 次数不超过(1)200,000，(2)25,000，(3)20,000。结果正确的是_____。

正确答案：C。解析：依据大关系算法复杂性“ $B_R B_S / (M-2) + B_S$ ”。如 $B_R B_S / (M-2) + B_S < 200,000$ ，则可求出 M 应大于 530，如 $B_R B_S / (M-2) + B_S < 25,000$ ，则可求出 M 应大于 6700。20,000 次只允许 R 和 S 各装入内存一次，此时只需要将 R 或 S 中的某一个完全装入内存，而另一个可以一块一块装入，即 M 应大于 10,000 即可。故选项 C 是正确的。

1907、下面是一个迭代器实现算法。该算法实现的是_____操作。

```
Open() {
    R.Open();
    CurRel := R;
}
GetNext() {
    IF ( CurRel == R ) {
        t:= R.GetNext();
        IF (t<> NotFound)
            RETURN t;
        ELSE {
            S.Open();
            CurRel := S;
        }
    }
    RETURN S.GetNext();
}
Close() {
    R.Close();
    S.Close();
}
```

正确答案：A。解析：此迭代器算法实现的是并运算；

1908、下面是一个迭代器实现算法。该算法实现的是_____操作。

```
Open() {
    R.Open(); S.Open();
    r:= R.GetNext();
}
GetNext() {
    REPEAT {
        s:= S.GetNext();
        IF ( s == NotFound ) {
            S.Close();
            Return r;
        }
        IF ( r == s )
        { r:= R.GetNext();
          S.Close(); S.Open();
          s := S.GetNext();
        }
    }
    UNTIL (r== NotFound);
}
Close() {
    R.Close(); S.Close();
}
```

正确答案：C。解析：此迭代器算法实现的是 R-S 运算；

1909、下面是一个迭代器实现算法。该算法实现的是_____操作。

```
Open() {
    R.Open(); S.Open();
    r:= R.GetNext();
}
GetNext() {
    REPEAT {
        s:= S.GetNext();
        IF ( s == NotFound ) {
            S.Close();
            r:= R.GetNext();
            IF ( r == NotFound )
                RETURN NotFound;
            ELSE { S.Open();
                  s := S.GetNext(); }
        }
    }
    UNTIL (r与s相同);
    RETURN r;
}
Close() {
    R.Close(); S.Close();
}
```

正确答案：B。解析：此迭代器算法实现的是 $R \cap S$ 运算；

1910、假设关系 R 的元组个数为 $T(R)$ ，元组的大小为 $I(R)$ ，存储块的大小为 b ， $B(R)=T(R)*I(R)/b$ 。关于表空间扫描算法，下列说法正确的是_____。

正确答案：**B**。解析：利用表空间扫描算法，还要看关系是如何存储的，如果是聚簇关系(一个块中仅是该关系的元组)，则其复杂性为 $B(R)$ ，但如果不是聚簇关系，如果一个块中多个关系的元组混合存放，则可能一个元组在一个块中，则其复杂性为 $T(R)$ 。故选项 **B** 是正确的。选项 **E**，因为其他中有正确的，故此不正确。

1911、已知关系 R 的参数如下：聚簇存储磁盘块数 $B(R) = 1,000$ ，元组数 $T(R) = 20,000$ ， R 中属性 A 的不同值的个数记为 $V(R, A)=100$ 。 R 上有基于属性 A 的排序索引。关于 $\sigma_{A=0}(R)$ ，下列说法正确的是_____。

正确答案：**A**。解析：选项 **A** 是正确的，聚簇存储且不使用索引，查询代价为 $B(R)$ 。选项 **B** 是不正确的，因为如果 R 是非聚簇存储的，则其执行代价可能是 $I(R)=20,000$ ，此时要看 R 是聚簇存储的还是非聚簇存储的。选项 **C** 是不正确的，如果使用索引，则其执行代价通常是远小于 $B(R)$ 的。选项 **D** 是不正确的，如果 R 是聚簇存储的且使用索引，则查询代价 $=B(R)/V(R, A)=1000/100=10$ 个 I/O。

1912、关于去重复 $\&(R)$ 操作的一趟扫描算法，下列说法不正确的是_____。

正确答案：**D**。解析：选项 **A** 说法是正确的，只要内存块数足以装得下整个关系，即可执行。更严格的说，只要内存块数足以装得下关系中不重复值的元组数目，即可执行。选项 **B** 说法是正确的。选项 **C** 说法是正确的，可以通过散列，将具有相同值的元组散列到同一内存块中，因此同一内存块中没有重复，则即是没有重复。选项 **D** 说法不正确，不需要首先对 R 的所有数据建立内存数据结构，可以边执行边建立即可。

1913、分组聚集操作的一趟扫描算法_____。

正确答案：**D**。解析：选项 **A** 说法是正确的，只要内存块数足以装得下整个关系，即可执行。更严格的说，只要内存块数足以装得下关系中所有的分组及其每一分组上的不同值，即可执行。选项 **B** 说法是正确的。选项 **C** 说法是正确的，可以通过散列，将具有相同分组值的元组散列到同一内存块中。选项 **D** 说法是不正确的，有些分组聚集计算是在建立起完整的数据结构后才能进行，但有些是可以的。选项 **E** 说法是正确的，需要首先对 R 的所有数据建立内存数据结构(将相同组的数据聚集在一起)，然后再分组聚集计算即可。