

一、单选题 (10 分, 每题 2 分)

1. D
2. B
3. B
4. A
5. C

二、填空题 (10 分, 每空 1 分)

1. 元组
2. 选择                  差
3. 联系
4. 部分                  传递
5.  $\Pi_{RA, RB, RC, RD, SE}(\sigma_{RC=SC \wedge RD=SD}(R \times S))$   
 $(\Pi_{RA, RB, RC, RD, SE}(\sigma_{RC=SC \wedge RD=SD}(R \times S))) \cup \{(null, null)\} \times (S - \Pi_{SC, SD, SE}(\sigma_{RC=SC \wedge RD=SD}(R \times S)))$

三、查询题 (20 分, 每题四分)

1.  $\Pi_{ENAME}(\sigma_{D.DNAME="Research" \wedge E.SALARY > 8000 \wedge E.D\# = D.D\#}(\rho_E(EMPLOYEE) \times \rho_D(DEPARTMENT)))$
2.  $\Pi_{E\#}(\sigma_{P\#="P1" \vee P\#="P2"}(WORKS\_ON))$
3. `Select DNAME, avg(SALARY) as AVGDSALARY From DEPARTMENT  
Group by D# ;`
4. `Select ENAME From EMPLOYEE  
Where E# not in (  
Select E# From WORKS_ON  
Where P#="P1"  
);`
5. `Select DISTINCT E#, ENAME From EMPLOYEE, PROJECT, WORKS_ON  
Where EMPLOYEE.E#=WORKS_ON.E# and  
PROJECT.P#=WORKS_ON.P# and  
EMPLOYEE.D#<>PROJECT.D#`

四、计算和证明题 (20 分, 每题五分)

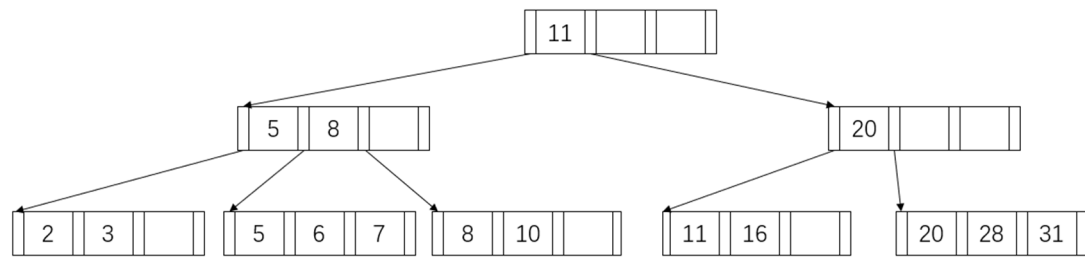
1. {ABCDE}
2. AB、BD
3. {A→E, A→D, D→A, BE→C}
4. {AB 或 BD, AE, AD, BEC}

五、设计题 (15 分)

言之有理即可

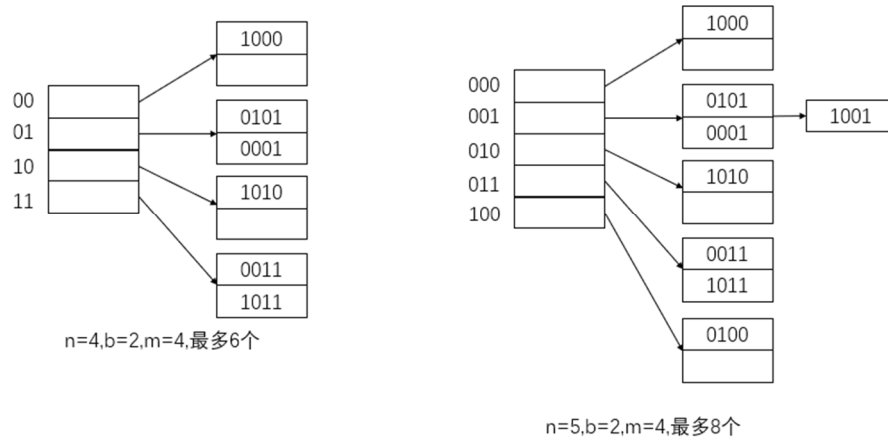
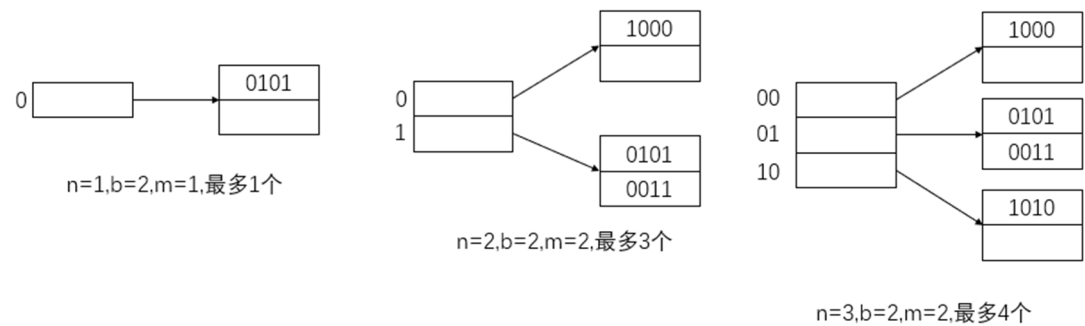
## 六、算法题（6分）

### 1.（3分）



### 2.（3分）

5: 0101      8: 1000      19: 0011      26: 1010      30: 1011  
33: 0001      36: 0100      41: 1001



## 七、分析题（6分）

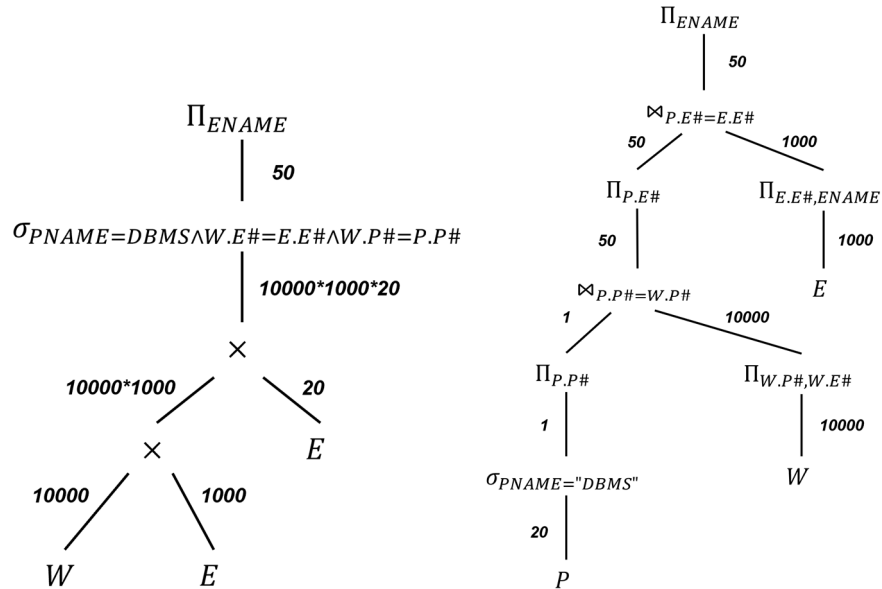
- （2分） $B(\text{Class})=10000/100=100$      $B(\text{Student})=5000/20=250$   
外关系是 Class 表，内关系是 Student 表。
- （2分） $B(\text{Class})+B(\text{Class}) \cdot B(\text{Student})/(M-1)=100+100 \cdot 250/10=100+2500=2600$
- （2分）不可以，因为  $B(\text{Class})+B(\text{Student})=100+250>11 \cdot 11=121=M \cdot M$ ，所以不满足可用内存页数的要求。

八、分析题 (6 分)

1. (2 分)

$$\Pi_{ENAME}(\sigma_{PNAME="DBMS" \wedge W.E\#=E.E\# \wedge W.P\#=P.P\#}(\rho_W(WORKS\_ON) \times \rho_P(PROJECT) \times \rho_E(EMPLOYEE)))$$

2. (4 分)



九、分析题 (6 分)

1. (2 分)

不是，优先图中存在环

$$T_0 \leftrightarrow T_2 \rightarrow T_1$$

2. (1) (2 分)

$T_1$ :  
 lock-x(A);  
 read(A, t);  
 t:=t+1;  
 write(A, t);  
 lock-x(B);  
 unlock(A);  
 read(B, t);  
 t:=t+1;  
 write(B, t);  
 unlock(B).

$T_2$ :  
 lock-x(A);  
 read(A, s);  
 s:=s\*2;  
 write(A, s);  
 lock-x(B);  
 unlock(A);

```
read(B, s);  
s:=s*2;  
write(B, s);  
unlock(B).
```

(2) (2 分)

不存在发生死锁的并发调度序列。一个事务会先释放掉对数据项的互斥锁，另一个事务才可以申请对该数据项的互斥锁，不存在相互等待的问题。

十、分析题 (6 分, 每空 1.5 分)

<T0, A, 100, 20>

<T2, START>

<T2, C, 60, -20 >

< CHECKPOINT T3>