南京理工大学

2017 年硕士学位研究生入学考试试题

科目代码：514　　科目名称：数据库与计算机网络　　满分：100 分

注意：①认真阅读答题纸上的注意事项；②所有答案必须写在答题纸上，写在本试题纸或草稿纸上均无效；③本试题纸须随答题纸一起装入试题袋中交回！

数据库

一、填空题（请在答题纸上标明序号，共 10 分）

1、（每空 1 分，共 6 分）假设关系 R（A,B）和关系 S（B,C）分别有 r 个和 s 个元组，且 r>0，s>0，对于下面的表达式，填写结果中可能出现的最多和最少的元组数目。

表达式	最多的元组数目	最少的元组数目
$RU\rho_{s(A,B)}S$	①	②
$\Pi_{AC}(R\infty S)$	③	④
$\Pi_{B}(R)-(\Pi_{B}(R)-\Pi n(S))$	⑤	⑥

2、（2 分）考虑以下关系声明：

CREATE TABLE R (a INT，b INT，c INT，

CHECK (3 < (SELECT AVG (b)　FROM R)));

R 已包含元组（1，4，14），（2，3，15）和（3，3，16），若试图执行语句

DELETE FROM R WHERE a=1；

该语句将被_____（允许、拒绝、置空或级联）执行。

3、（2 分）考虑以下关系声明：

CREATE TABLE Emps (id INT，ssNo INT，name CHAR(20)，managerID INT)；若将该声明进行扩展，使得 id 和 ssNo 均为 key（键），managerID 分量值必须出现在 id 属性的分量中，请在下划线处填空，完成对声明的扩展。

CREATE TABLE Emps （_____）；

二、设计题（共 20 分）

1、（10 分）考虑关系

R(student, name, course, grade, department, dean)，其中属性依次表示学生学号、学生姓名、课程编号、成绩、院系、主任。规定：

☆学号唯一标识学生，课程编号表意标识课程；

☆每名学生只有一个姓名，只能所属一个院系。

☆每个院系只有一位主任。

☆每名学生可以选修多门课程；

☆每名学生在每门课程上只有一个成绩。

根据以上描述，

1）给出关系 R 上的函数依赖。（3 分）

2）找出关系 R 的所有键。（2 分）

3）请问 R 最高满足第几范式？（2 分）
4）若 R 不属于 3NF，请将 R 分解，使分解后的关系均满足 3NF。（3 分）
2、（10 分）为某医院管理系统设计数据库，需要管理：
☆每名病人有唯一的身份证号、姓名和一家保险公司；
☆每名医生有唯一的工号、姓名、专业和办公室电话；
☆每名护士有唯一的工号、姓名、所属护士站编号和办公室电话；
☆每名医院管理人员有唯一的工号、姓名、所在管理部门和办公室电话；
☆医生、护士和管理人员均属于医院的职工；
☆每位病人在一次医疗服务中对应一名医生和若干名护士；
☆医生、护士都能为多次医疗提供服务，病人也可接受多次医疗服务。
针对以上要求，
1）请完成 E/R 图设计。注意正确地表示实体、属性、联系多重性和关键字。（5 分）
2）请根据 E/R 图设计关系模式。注意正确表示属性和关键字。（5 分）

三、编程题（共 20 分）

考虑一个大学篮球管理数据库的部分关系；
Player (playerID:int, Name: string, Number:int, TeamID:int,
Position: string)
Team (TeamID:int， School: string, Mascot: string, Coloro:otring)
Stats (Year:int, PlayerID:int, Points:int, FieldGoalPercent: int,
3PntFGP:int, Rebounds:int, Turnovers:int)
其中，Player 记录篮球运动员个人信息，依次为运动员编号、姓名、球衣号、所在球队编号和位置（后卫、前锋、中锋等）；Team 记录球队信息，依次为球队编号、所属大学、吉祥物和队服颜色（绿色、黑色和橘色、黄色和绿色，等）；Stats 记录运动员每年统计信息，依次为年份、运动员编号、得分、投篮命中率、三分球命中率、篮板球数和失误数。
回答以下问题：
1、基于以上关系模式，用关系代数语言写出下列查询。（每小题 2 分，共 6 分）
1）列出“李华”所在大学。
2）列出不是“后卫”的所有队员的名字。
3）列出 2016 年投篮命中率最高值。
2、分别用单条 SQL 语言完成下列查询。（不允许出现 MINUS、INTERSECT、EXCEPT 关键字）（1、2 小题各 2 分，3、4 小题各 3 分，共 10 分）
1）查询队服包含“黑色”的球队编号、所属大学和队服颜色。
2）查询年度投篮命中率从未低于 50 的运动员编号、姓名和所在球队。
3）查询 2016 年投篮命中率最高的运动员编号、姓名和所在球队。
4）针对每个球队，查询旗下运动员 2016 年平均得分和投篮命中率。
3、用 SQL 语言完成下列操作。（每小题 2 分，共 4 分）
1）编号为 123 的运动员转学后，所在球队编号变更为 16，球衣号未定。
2）增加 2016 年 124 号运动员的统计信息：得分 359，投篮命中率 44.6，投篮命中率为 39.7，篮板球数 243，失误数 67。

四、名词解释（共 10 小题，每小题 2 分，共 20 分）

1、拥塞控制　　2、PDU　　3、内容分布网络
4、无连接服务　　5、生成树　　6、毒性逆转

7、自治系统

9、层次选路（Hierarchical Routing）　　10、RTP

五、简答题（共 5 题，每小题 4 分，共 20 分）

1、什么是 Socket？什么是 SAP？试对这两个概念进行比较。

2、简述 CSMA/CD 与 CSMA/CA 有什么不同？无线网络为什么用 CSMA/CA？

3、按照五层网络协议模型，试说明两个系统用户进程层之间的通信过程。

4、简述应用级网关的主要功能是什么？应用级网关与分组过滤有什么不同？

5、按照如下公式计算往返延时与超时，试解释计算过程是如何实现的？这样计算得到的 TimooutInterval 合理吗？

EstimatedRTT=(1-a)×EstimatedRTT+a×SampleRTT　　(1)

DevRTT-(1-β)×DevRTT+β×(SampleRTT-EstimatedRTT)　　(2)

TimeoutInterval=EstimatedRTT+4×DevRTT　　(3)

a 参考值为 0.125，β 的推荐值为 0.25。

六、问答题（共 2 题，每题 5 分，共 10 分）

1、如图 1 的网络，按照标明的链路费用，用 Dijkstra 算法计算出从 C 节点到所有网络节点的最短路径，并画出最短通路树，并给出路由表：如果其他网格节点不用 Dijkstra 算法计算最短路径，而采用 C 节点计算出来的最短通路树，假设 C 节点将它计算出来的最短通路树发给其他节点，试说明这样做的优点和缺点。

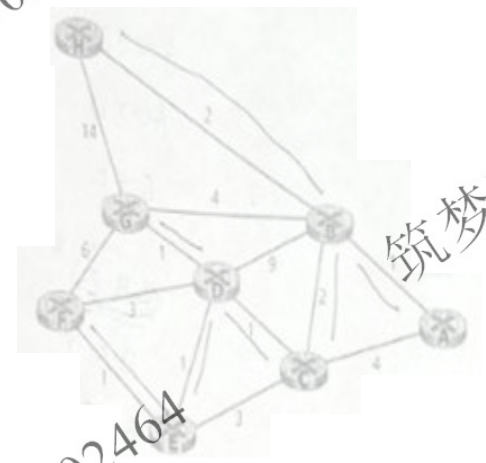

图 1；

2、图 2 是 rdt3.0 协议中发送方和接收方的有限状态图（FSM），指出 rdt3.0 sender 以及 rdt3.0 receiver 算法中存在的问题？

Rdt3.0 sender

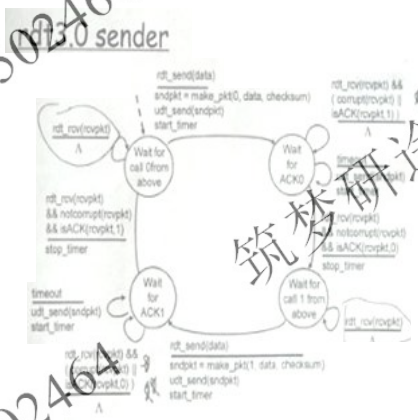

图 2；

南京理工大学 2017 年硕士学位研究生入学机试试题

（满分 120 分）

第 1 题：连续字母（标号：A，分值：20）

（时间限制：1000ms，内存限制 65536 KB）

题目描述

给你一个字符串，只包含大写字母，求同一字母连续出现的最大次数，例如“AAAABBCDHHH”同一字母连续出现的最大次数为 4，因为一开始 A 连续出现了 4 次。

输入描述

每组输入第一行有一个整数 cases(1<=cases<=100)，表示有 cases 个测试数据。接下来每行有一个字串（1<长度<=100）。

输出描述

输出对应每个子串同一字母连续出现的最大次数。

输入格式样例

3

AAAABBCDHHH

ISDHPSHFDAASDIAHSD

EEEEEEE

输出格式样例

4

2

第 2 题：疯狂的快递哥（标号：B，分值：20）

（时间限制：1000ms，内存限制 65536 KB）

题目描述

在一个非常景气的企业内，有一名疯狂的快递哥，做什么？当然是送快递了！一天，他有一件快递要送到城市 n，而他的 gps 坏了，不知道如何以最快的速度到达那里，就来寻求你的帮忙，该企业位于城市 1，你能在给定的地图中，替他安排路径，使得他能以最快速度到达目的地吗？（快递小哥行进速度不变）

输入描述

输入第一行含有两个数字 n 和 m，表明地图中城市数目和城市之间的路径的数目。城市 1 为快递小哥出发点，城市 n 为快递小哥的目的地。

接下来 m 行每行含有 3 个整数 a,b,c,表示城市 a 和城市 b 之间存在一条长为 c 的双向路径，两个城市间可能存在多条路径。

（0<n<100, 0<m<10000, 0<c<1000000000）

输出描述

按行走顺序输出快递小哥所行走经过的城市，城市编号间以一个空格间隔，最后没有多余的空格，若无法到达城市 n，则输出-1。（包括城市 1 和城市 n）

输入格式样例

```
5 6
1 2 1
2
3 4 5
3 5 10
1 3 2
4 5 4
```

输出格式样例

```
1 3 4 5
```

第 3 题：subrange_sum（标号：C，分值：20）

（时间限制：2000 ms，内在限制 65536 KB）

题目描述

给一个含有 c 个整数的数组，求出有多少个下标连续的区间，使得区间内所有数字的和大于等于 x。

输入描述

第一行两个整数 c x(0 < c <= 1000000, -100000000 <= x <= 100000000) 第二行有 c 个整数（每个数字的绝对值小于等于 100，数字之间用空格隔开）。

输出描述

输出一个整数，表示所求的个数。

输入格式样例

```
3 2
2 -4 7
```

输出格式样例

```
4
```

对于有 3 个整数构成的数组而言，总共有 6 个下标连续的区间，他们的和分别为：2=2-4=-4 7=72+-4=-2-4+7=32+-4+7=5 其中有 4 个和大于等于 2，所以答案等于 4。

第 4 题：进制转化（标号：D，分值：20）

（时间限制：1000 ms，内存限制 65536 KB）

题目描述

给定一个十进制正整数，要求输出这个正整数的二进制数、八进制数和十六进制数。

输入描述

输入数据只有一行，包含一个十进制正整数 x（1 <= x <= 1000000000）

输出描述

输出一共 3 行，第一行输出二进制数，第二行输出八进制数，第三行输出十六进制数，十六进制数中的字母采用大写字母。

输入格式样例
13

输出格式样例
1101
15
D
提示
十六进制中，A=10，B=11，C=12，D=13，E=14，F=15

第 5 题：疯狂的修路哥（标号：E，分值：20）
（时间限制：1000 ms，内存限制 65536 KB）

题目描述
在一个发展中的城市，由于地区的分布散乱，交通极不方便，政府打算修建一些路使这些地区两两之间直接连接或间接连接，现政府想请你帮忙编写一个程序来解决这个问题，为了简化问题，这些地区被编号为 1~n，而路的修建代价即为其长度。

输入描述
输入第一行包含一个数字 n，代表城市的数量。
接下来的第 i 行含每行包含 2 个整数数字 xi 和 yi，为城市 i 平面坐标。
（0<n<100, 0<=xi, yi<=100000）

输出描述
输出仅含一个数字，即为修建路所花费的代价，小数点都保留两位小数。

输入格式样例
4
0 0
100 0
0 1
1 100

输出格式样例
102.00

第 6 题：最长回文子序列（标号：F，分值：20）
（时间限制：1000 ms，内存限制 65536 KB）

题目描述
给定一仅由大小写字母组成的字符串，求其回文子序列的最大长度。回文子序列指正着反着读都一样的子序列，如 madam。本题中大小写不敏感，即 a 和 A 可以看做是一样的。

输入描述
输入包含一行字符串，其长度不超过 300。

输出描述
输出该字符串最长回文子序列的长度。

输入格式样例

ABCdca

输出格式样例

5

提示

子序列不要求连续。

南京理工大学

2017 年硕士学位研究生入学复试上机试题

（参考答案）

第一题：

```
#include<iostream>
#include <string>
using namespace std;

int main()
{
    int n;
    int count[100];
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        string s;
        cin >> s;
        count[i] = 1;
        int countt = 1;
        for (int j = 0; j < s.length(); j++)//从第一个开始比较，相同的加 1，然后和原来存的次数比较，小于不改变，大于改变，并重新计数
        {
            if (s[j] == s[j+1])
                countt++;
            else
            {
                if (countt > count[i])
                {
                    count[i] = countt;
                    countt = 0;
                }
            }
        }
    }
    for (int i = 0; i < n; i++)
        cout << count[i]<<endl;

    return 0;
}
```

第二题：

5 6
1 2 1
2 3 4
3 4 3
3 5 10
1 3 2
4 5 4

求最短路径问题

答案：

第三题：

```
#include<iostream>
using namespace std;

int main()
{
    int c, x;
    cin >> c >> x;
    int a[10000];
    for (int i = 0; i < c; i++)
    {
        int s;
        cin >> s;
        a[i] = s;
    }
    int count = 0;//大于等于的个数
    int bijiao=0;//当前比较的数字和
    for (int i = 1; i <= c; i++)//比较长度从 1 到 c，比较次数从 c 到 1
    {
        for (int j = 1; j<=c-i+1; j++) //执行比较次数
        {
            for(int t=0;t<i;t++)//从 0 开始比较，长度为 i
            {
                bijiao = bijiao + a[j-1];
            }
            if (bijiao >= x)
                count++;

            bijiao = 0;
        }
    }
```

```
    cout << count << endl;
    return 0;

}、
```

第四题：

十进制整数转换为二进制整数采用"除 2 取余，逆序排列"法。

具体做法是：用 2 整除十进制整数，可以得到一个商和余数；再用 2 去除商，又会得到一个商和余数，如此进行，直到商为 0 时为止，

然后把先得到的余数作为二进制数的低位有效位，后得到的余数作为二进制数的高位有效位，依次排列起来。

```
//进制转化
#include<iostream>
using namespace std;
void main()

    int n, i, j = 0;
    int a[1000];//存储 2 进制编码
    cin >> n;
    i = n;
    //转换为 2 进制
    while (i)//对 2 取余并除 2，直到商为 0 时为止
    {
        a[j] = i % 2;
        i /= 2;
        j++;
    }
    for (i = j - 1; i >= 0; i--)//逆序输出
        cout << a[i] ;
    cout << endl;
    //转换为 8 进制
    i = n,j=0;
    while (i)//对 8 取余并除 8，直到商为 0 时为止
    {
        a[j] = i % 8;
        i /= 8;
        j++;
    }
    for (i = j - 1; i >= 0; i--)//逆序输出
        cout << a[i];
    cout << endl;
    //转换为 16 进制
    i = n, j = 0;
    while (i)//对 16 取余并除 16，直到商为 0 时为止
```

```
    {
        a[j] = i % 16;
        i /= 16;
        j++;
    }
    for (i = j - 1; i >= 0; i--)//逆序输出    10 为 A 11 B 12C 13 D 14 E 15F
    {
        if (a[i] == 10)
            cout << 'A';
        else
            if (a[i] == 11)
                cout << 'B';
            else
                if (a[i] == 12)
                    cout << 'C';
                else
                    if (a[i] == 13)
                        cout << 'D';
                    else
                        if (a[i] == 14)
                            cout << 'E';
                        else
                            if (a[i] == 15)
                                cout << 'F';
                            else
                                cout << a[i];
    }
}
```

第五题：

求图的最小生成树

求图的最小生成树主要有两种经典算法：

1.普里姆算法

时间复杂度为 O(n2).适合于求边稠密的最小生成树。

2.克鲁斯卡尔算法

时间复杂度为 O(eloge)(e 为网中边数），适合于求稀疏的网的最小生成树。

第六题：

```
#include <string>
#include<iostream>
using namespace std;
string findLongestPalindrome(string &s);
void main()
```

```
{
    //首先将大写字母都变成小写字母
    string s ;
    cin >> s;
    int len = s.size();
    for (int i = 0; i<len; i++)
    {
        if (s[i] >= 'A' && s[i] <= 'Z')
        {
            s[i] += ('a' - 'A');
        }
    }
    cout << findLongestPalindrome(s).length();
}
string findLongestPalindrome(string &s)    //求回文字符串

    int length = s.size();//字符串长度
    int maxlength = 0;//最长回文字符串长度
    int start;//最长回文字符串起始地址
    for (int i = 0; i<length; i++)//起始地址
        for (int j = i + 1; j<length; j++)//结束地址
        {
            int tmp1, tmp2;
            for (tmp1 = i, tmp2 = j; tmp1<tmp2; tmp1++, tmp2--)//判断是不是回文
            {
                if (s.at(tmp1) != s.at(tmp2))
                    break;
            }
            if (tmp1 >= tmp2&&j - i>maxlength)
            {
                maxlength = j - i + 1;
                start = i;
            }
        }
    if (maxlength>0)
        return s.substr(start, maxlength);//求子串
    return NULL;
```