第七届（2001）分区联赛复赛解题报告（提高组）

第一题：一元三次方程求解(p1.pas p1.in p1.out)

问题描述

有形如：$ax^3+bx^2+cx+d=0$　这样的一个一元三次方程。给出该方程中各项的系数(a，b，c，d　均为实数)，并约定该方程存在三个不同实根(根的范围在-100至100之间)，且根与根之差的绝对值>=1。要求由小到大依次在同一行输出这三个实根(根与根之间留有空格)，并精确到小数点后2位。提示：记方程f(x)=0，若存在2个数x1和x2，且x1<x2，f(x1)*f(x2)<0，则在(x1，x2)之间一定有一个 根。

样例

输入：1　　-5　　-4　　20

输出：-2.00　　2.00　　5.00

第二题：数的划分(p2.pas/c/cpp p2.in p2.out)

问题描述

将整数n分成k份，且每份不能为空，任意两份不能相同(不考虑顺序)。例如：n=7，k=3，下面三种分法被认为是相同的。1，1，5; 1，5，1; 5，1，1;问有多少种不同的分法。

输入：n，k (6<n<=200，2<=k<=6)

输出：一个整数，即不同的分法。

样例

输入： 7 3

输出：4 {四种分法为：1，1，5;1，2，4;1，3，3;2，2，3;}

第三题：统计单词个数(p3.pas/c/cpp　p3.in　p3.out)

问题描述

给出一个长度不超过200的由小写英文字母组成的字母串(约定;该字串以每行20个字母的方式输入，且保证每行一定为20个)。要求将此字母串分成k份(1<k<=40)，且每份中包含的单词个数加起来总数最大(每份中包含的单词可以部分重叠。当选用一个单词之后，其第一个字母不能再用。例如字符串this中可包含this和is，选用this之后就不能包含th)。单词在给出的一个不超过6个单词的字典中。

要求输出最大的个数。

输入格式:输入数据放在文本文件p3.in中，其格式如下：第一行为一个正整数(0<n<=5)表示有n组测试数据每组的第一行有二个正整数(p，k),p表示字串的行数;k表示分为k个部分。接下来的p行，每行均有20个字符。再接下来有一个正整数s，表示字典中单词个数。(1<=s<=6)接下来的s行，每行均有一个单词。

输出格式:结果输出至屏幕，每行一个整数，分别对应每组测试数据的相应结果。

样例

输入：

```
1
1 3
thisisabookyouareaoh
4
is
a
ok
sab
```

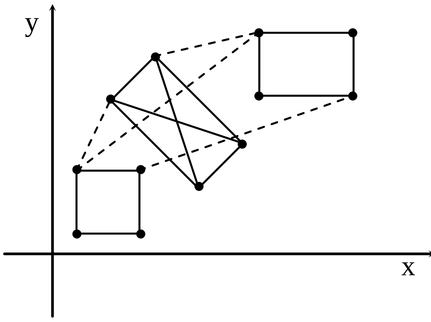
输出： //说明：(不必输出)

```
7 // this/isabookyoua/reaoh
```

第四题：CAR 的旅行路线(p4.pas/c/cpp　p4.in　p4.out)

问题描述

又到暑假了，住在城市 A 的 Car 想和朋友一起去城市 B 旅游。她知道每个城市都有四个飞机场，分别位于一个矩形的四个顶点上，同一个城市中两个机场之间有一条笔直的高速铁路，第 I 个城市中高速铁路了的单位里程价格为 Ti，任意两个不同城市的机场之间均有航线，所有航线单位里程的价格均为 t。



图例：

・机场

—高速铁路

-- 飞机航线

注意：

图中并没有标出所有的铁路与航线。

那么 Car 应如何安排到城市 B 的路线才能尽可能的节省花费呢?她发现这并不是一个简单的问题，于是她来向你请教。

任务:找出一条从城市 A 到 B 的旅游路线，出发和到达城市中的机场可以任意选取，要求总的花费最少。

输入文件：输入文件名 p4.in

输 出：p4.out(输出最小费用，小数点后保留 1 位。)

输入格式:第一行为一个正整数 n(0<=n<=10)，表示有 n 组测试数据。每组的第一行有四个正整数 s，t，A，B。S(0<S<=100)表示城市的个数，t 表示飞机单位里程的价格，A，B 分别为城市 A，B 的序号，(1<=A，B<=S)。接下来有 S 行，其中第 I 行均有 7 个正整数 xi1，yi1，xi2，yi2，xi3，yi3，Ti，这当中的(xi1，yi1)，(xi2，yi2)，(xi3，yi3)分别是第 I 个城市中任意三个机场的坐标，T I 为第 I 个城市高速铁路单位里程的价格。

输出格式:共有 n 行，每行一个数据对应测试数据。

样例

输入

```
1
1 10 1 3
1 1 1 3 3 1 30
2 5 7 4 5 2 1
8 6 8 8 11 6 3
```

输出：

```
47.55
```

参考程序

第一题：

```
var
  y,n,a,b,c,d:real;
  i:longint;

begin
  assign(input,’p1.in’);reset(input);
  assign(output,’p1.out’);rewrite(output);
  read(a,b,c,d);
  for i:=-10000 to 10000 do
    begin
      n:=i/100;
      y:=n*n*n*a+n*n*b+n*c+d;
      if abs(y)<1e-5 then write(n:0:2,’ ’);
    end;
  close(input);close(output);
end.
```

第二题：

```
var
  n,k:longint;

function f(n,k,s:longint):longint;
var
  sum,i:longint;
begin
```

```
  sum:=0;
  if k=1 then exit(1);
  for i:=s to n div k do
    sum:=sum+f(n-i,k-1,i);
  exit(sum);
end;

begin
  assign(input,'p2.in');reset(input);
  assign(output,'p2.out');rewrite(output);
  readln(n,k);
  write(f(n,k,1));
  close(input);close(output);
end.
```

第三题：

```
Var
  p,k,s,n:integer;
  ss:string[200];
  word:array[1..6]of string[200];
  g:array[1..200,1..200]of byte;
  f:array[1..200,1..40]of integer;
  w:array[1..200]of byte;

function max(a,b:integer):integer;
begin
  if a>b then exit(a);
  exit(b);
end;

procedure inIt;
var
  i,j:integer;
  s1:string;
begin
  ss:='';
  readln(p,k);
  for i:=1 to p do
    begin
      readln(s1);
      ss:=ss+s1;
    end;
  n:= 20*p;
  readln(s);
  for i:=1 to s do readln(word[i]);
  fillchar(g,sizeof(g),0);
  fillchar(f,sizeof(f),0);
  for i:=1 to n do
    begin
      w[i]:=200;
      for j:=1 to s do
        if (copy(ss,i,length(word[J]))=word[j])and(length(word[j])<w[i])then
w[i] := length(word[j]);
    end;
  for i:=1 to n do
    for j:=i to n do
      for p:=i to j do
        if (w[p]+p-1<=j) then g[i,j]:=g[i,j]+1;
end;

procedure solve;
var
  i,j:integer;
begin
  for i:=1 to n do
    f[i,1]:=g[1,i];
  for p:=2to k do
    for i:=p to n do
      for j:=p-1 to i-1 do
        f[i,p]:=max(f[i,p],f[j,p-1]+g[j+1,i]);
  writeln(f[n,k]);
end;

begin
  assign(input,’p3.in’);reset(input);
  assign(output,’p3.out’);rewrite(output);
  init;
  solve;
  close(input);close(output);
end.
```

第四题：

方法一：

Folyd：

```
const
  inf=1e+38;
var
  s,a,b,m,n,t:longint;
  tr:array[1..100]of longint;
  x,y:array[1..100,1..4]of longint;
  w:array[0..400,0..400]of real;

function dis(x1,y1,x2,y2:longint):real;
begin
  dis:=sqrt(sqr(x2-x1)+sqr(y2-y1));
end;

procedure solve;
var
  i,j,k,i1,j1:longint;
  l:real;
begin
  readln(s,t,a,b);
  if a=b then
    begin
      writeln(’0.00’);
      writeln;
      exit;
    end;
  for m:=1 to s do
    begin
      readln(x[m,1],y[m,1],x[m,2],y[m,2],x[m,3],y[m,3],tr[m]);
      for i:=1 to 2 do
        for j:=1 to 3 do
          if i<>j then
            if
(x[m,j]-x[m,i])*(x[m,6-i-j]-x[m,j])+(y[m,j]-y[m,i])*(y[m,6-i-j]-y[m,j])=0 then
              begin
                x[m,4]:=x[m,i]-x[m,j]+x[m,6-i-j];
                y[m,4]:=y[m,i]-y[m,j]+y[m,6-i-j];
              end;
    end;
  for i:=1 to s do
    for j:=1 to 4 do
      for k:=1 to 4 do
        w[4*i-4+j,4*i-4+k]:=dis(x[i,j],y[i,j],x[i,k],y[i,k])*tr[i];
  for i:=1 to s do
    for i1:=1 to 4 do
```

```
      for j:=1 to s do
        if i<>j then
          for j1:=1 to 4 do
            w[4*i-4+i1,4*j-4+j1]:=dis(x[i,i1],y[i,i1],x[j,j1],y[j,j1])*t;

  for k:=1 to 4*s do
    for i:=1 to 4*s do
      if i<>k then
        for j:=1 to 4*s do
          if (w[i,j]>w[i,k]+w[k,j])and(j<>k)and(j<>i)then w[i,j]:=w[i,k]+w[k,j];

  l:=1e38;
  for i:=1 to 4 do
    for j:=1 to 4 do
      if w[a*4-4+i,b*4-4+j]<l then l:=w[a*4-4+i,b*4-4+j];
  writeln(l:0:2);
  writeln;
end;

begin
  assign(input,'car.in');reset(input);
  assign(output,'car.out');rewrite(output);
  readln(n);
  for t:=1 to n do solve;
  close(input);close(output);
end.
```

方法二：

单元最短路径：

```
const
  inf=1e+38;
var
  s,a,b,m,n,t:longint;
  tr:array[1..100]of longint;
  x,y:array[1..100,1..4]of longint;
  z:array[0..100,1..4,1..4]of real;
  f:array[0..100,1..4,0..100,1..4]of real;
  w:array[0..100,1..4]of real;

function dis(x1,y1,x2,y2:longint):real;
begin
  dis:=sqrt(sqr(x2-x1)+sqr(y2-y1));
```

```
end;

function dij(c:longint):real;
var
  v:array[1..100,1..4]of boolean;
  i,j,k,g,mc,mp:longint;
  min:real;
begin
   fillchar(v,sizeof(v),false);
   for i:=1 to s do
     for j:=1 to 4 do
       w[i,j]:=f[a,c,i,j];
   w[a,c]:=0;
   for i:=1 to 4 do v[a,i]:=true;
   mc:=a; mp:=c;
   for j:=1 to 4*s-8 do
     begin
       min:=inf;
       for i:=1 to s do
         for k:=1 to 4 do
           if (not v[i,k])and(w[i,k]<min) then
             begin
               mc:=i;
               mp:=k;
               min:=w[i,k];
             end;
       v[mc,mp]:=true;
       if mc=b then continue;
       for i:=1 to s do
         for k:=1 to 4 do
           for g:=1 to 4 do
             if (w[i,k]>w[mc,mp]+f[mc,g,i,k]+z[mc,mp,g])then
w[i,k]:=w[mc,mp]+f[mc,g,i,k]+z[mc,mp,g];
     end;
  min:=w[b,1];
  for i:=2 to 4 do
    if w[b,i]<min then min:=w[b,i];
  exit(min);
end;

procedure solve;
var
  i,j,i1,j1:longint;
  l,k:real;
```

```
begin
  readln(s,t,a,b);
  for m:=1 to s do
    begin
      readln(x[m,1],y[m,1],x[m,2],y[m,2],x[m,3],y[m,3],tr[m]);
      for i:=1 to 2 do
        for j:=1 to 3 do
          if i<>j then
            if
(x[m,j]-x[m,i])*(x[m,6-i-j]-x[m,j])+(y[m,j]-y[m,i])*(y[m,6-i-j]-y[m,j])=0 then
              begin
                x[m,4]:=x[m,i]-x[m,j]+x[m,6-i-j];
                y[m,4]:=y[m,i]-y[m,j]+y[m,6-i-j];
              end;
      for i:=1 to 3 do
        for j:=1 to 4 do
          if i<>j then
            begin
              z[m,i,j]:=dis(x[m,i],y[m,i],x[m,j],y[m,j])*tr[m];
              z[m,j,i]:=z[m,i,j];
            end
          else
            begin
              z[m,i,j]:=inf;
              z[m,j,i]:=inf;
            end;
    end;
  for i:=1 to s do
    for i1:=1 to 4 do
      for j:=1 to s do
        for j1:=1 to 4 do
          if i<>j then
            begin
              f[i,i1,j,j1]:=dis(x[i,i1],y[i,i1],x[j,j1],y[j,j1])*t;
              f[j,j1,i,i1]:=f[i,i1,j,j1];
            end
          else
            begin
              f[i,i1,j,j1]:=inf;
              f[j,j1,i,i1]:=inf;
            end;
  if a=b then
    begin
      writeln(’0.00’);
      writeln;
    end
  else
    begin
      l:=1e38;
      for i:=1 to 4 do
        begin
          k:=dij(i);
          if k<l then l:=k;
        end;
      writeln(l:0:2);
      writeln;
    end;
end;

begin
  assign(input,'car.in');reset(input);
  assign(output,'car.out');rewrite(output);
  readln(n);
  for t:=1 to n do solve;
  close(input);close(output);
end.
```

方法三：

单元最短路径（不是从 1 出发）：

```
const
  inf=1e38;
var
  tt,n,s,t,a,b,m:integer;
  tr:array[1..100]of longint;
  x,y:array[1..100,1..4]of longint;
  w,z:array[0..401,0..401]of real;
  f:array[0..401] of real;
  max:extended;

function dist(x1,y1,x2,y2:longint):real;
begin
  dist:=sqrt(sqr(x2-x1)+sqr(y2-y1));
end;

function diska(a:longint):real;
var
```

```
  v:array[0..401] of boolean;
  k,mc,mp,i,j,p:longint;
  max:real;
begin
  fillchar(v,sizeof(v),false);
  v[a]:=true;
  for i:=1 to s do
    for j:=1 to 4 do
       f[4*i-4+j]:=w[a,4*i-4+j];
  for j:=2 to 4*s do
    begin
      max:=inf; p:=a;
      for i:=1 to s do
        for k:=1 to 4 do
        if (f[4*i-4+k]<max)and(not v[4*i-4+k]) then
            begin
              max:=f[4*i-4+k];
              mc:=i;
              mp:=k;
            end;
      p:=4*mc-4+mp;
      v[p]:=true;
      for i:=1 to s do
        for k:=1 to 4 do
          if f[4*i-4+k]>f[p]+w[p,4*i-4+k] then f[4*i-4+k]:=f[p]+w[p,4*i-4+k];
    end;
  max:=inf;
  for i:=1 to 4 do
    if max>f[b*4-4+i] then max:=f[b*4-4+i];
  exit(max);
end;

procedure solve;
var
  i,j,k,l:longint;
  ans,min:real;
begin
  readln(s,t,a,b);
  if a=b then
    begin
      writeln('0.00');
      writeln;
      exit;
    end;
  for m:=1 to s do
    begin
      readln(x[m,1],y[m,1],x[m,2],y[m,2],x[m,3],y[m,3],tr[m]);
      for i:=1 to 2 do
        for j:=1 to 3 do
          if i<>j then
            if
(x[m,j]-x[m,i])*(x[m,6-i-j]-x[m,j])+(y[m,j]-y[m,i])*(y[m,6-i-j]-y[m,j])=0 then
              begin
                x[m,4]:=x[m,i]-x[m,j]+x[m,6-i-j];
                y[m,4]:=y[m,i]-y[m,j]+y[m,6-i-j];
              end;
    end;
  for i:=1 to s do
    for j:=1 to 4 do
      for k:=1 to 4 do
        w[4*i-4+j,4*i-4+k]:=dist(x[i,j],y[i,j],x[i,k],y[i,k])*tr[i];
  for i:=1 to s do
    for j:=1 to 4 do
      for k:=1 to s do
        if i<>k then
        for l:=1 to 4 do
          w[4*i-4+j,4*k-4+l]:=dist(x[i,j],y[i,j],x[k,l],y[k,l])*t;
  ans:=inf;
  for k:=1 to 4 do
    begin
      min:=diska(4*a-4+k);
      if min<ans then ans:=min;
    end;
  writeln(ans:0:2);
  writeln;
end;

begin
  assign(input,'car.in');reset(input);
  assign(output,'car.out');rewrite(output);
  readln(n);
  for tt:=1 to n do solve;
  close(input);  close(output);
end.
```