

Noip2011 解题报告

1.题意：将一个数数字位颠倒，符号不变。

注意：由题意，不会出现-0情况。

算法：模拟。

时间复杂度： $O(n \log_{10})$

空间复杂度： $O(1)$

代码：

```
Program reverse;
var
  n:longint;

procedure init;
begin
  assign(input,'reverse.in');
  assign(output,'reverse.out');
  reset(input);
  rewrite(output);
end;

procedure clo;
begin
  close(input);
  close(output);
end;

begin
  init;
  readln(n);
  if n<0 then
    begin
      write('-');
      n:=abs(n);
    end;
  while (n mod 10=0)and(n>0) do
    n:=n div 10;
  while n>0 do
    begin
      write(n mod 10);
      n:=n div 10;
    end;
  writeln;
  clo;
End.
```

测试情况：

题号	时间(s)	题号	时间(s)
1	0.05	6	0.04
2	0.03	7	0.03
3	0.03	8	0.04
4	0.05	9	0.03
5	0.04	10	0.05

2.题意：在一个字符串中寻找单词

注意：单词过长，用 ansistring 操作更简单，空间更小。

算法：

(1) 从前向后搜，

时间： $O(n^2)$

空间： $O(1)$

代码：

Program stat;

var

s1:ansistring;

s,s2:string;

i,num,first,i1:longint;

procedure init;

begin

assign(input,'stat.in');

assign(output,'stat.out');

reset(input);

rewrite(output);

end;

procedure clo;

begin

close(input);

close(output);

end;

begin

init;

readln(s);

for i:=1 to length(s) do

if (s[i]>='A')and(s[i]<='Z') then

s[i]:=chr(ord(s[i])-ord('A')+ord('a'));

readln(s1);

```

for i:=1 to length(s1) do
  if (s1[i]>='A')and(s1[i]<='Z') then
    s1[i]:=chr(ord(s1[i])-ord('A')+ord('a'));{大小写}
s1:=s1+' ';{方便以后操作}
i:=0;
first:=0;
num:=0;
while i<length(s1) do
  begin
    inc(i);
    while (s1[i]=' ')and(i<length(s1)) do inc(i);
    s2:="";
    i1:=i;
    while s1[i]<>' ' do
      begin
        s2:=s2+s1[i];
        inc(i);
      end;
    if s2=s then
      begin
        inc(num);
        if first=0 then first:=i1;
      end;
    end;{从前向后搜}
if num=0 then writeln('-1')
else
  writeln(num,' ',first-1);
clo;
end.

```

优化：本题数据量较小，优化与不优化差别不大。具体差别只有在对字符串处理。

测试情况：

题号	时间(s)	题号	时间(s)
1	0.03	6	0.04
2	0.03	7	0.03
3	0.03	8	0.17
4	0.05	9	0.17
5	0.04	10	0.17

3.题意：2*N 名编号为 1~2N 的选手共进行 R 轮比赛。每轮比赛开始前，以及所有比赛结束后，都会按照总分从高到低对选手进行一次排名。选手的总分为第一轮开始前的初始分数加上已参加过的所有比赛的得分和。总分相同的，约定编号较小的选手排名靠前。每轮比赛的对阵安排与该轮比赛开始前的排名有关：第 1 名和第 2 名、第 3 名和第 4 名、……、第 2K-1 名和第 2K 名、……、第 2N-1 名和第 2N 名，各进行一

场比赛。每场比赛胜者得 1 分，负者得 0 分。也就是说除了首轮以外，其它轮比赛的安排均不能事先确定，而是要取决于选手在之前比赛中的表现。

算法：模拟

(1) 每轮，对所有数排序。可过5-6个点

时间效率： $O(n^2 \lg n / \lg 2)$

```
var
  a,b,c:array[0..200000] of longint;
  p,q,r,i,n,j:longint;

procedure init;
begin
  assign(input,'swiss.in');
  assign(output,'swiss.out');
  reset(input);
  rewrite(output);
end;

procedure clo;
begin
  close(input);
  close(output);
end;

procedure qsort(l,r:longint);
var i,j,p,q,k,w:longint;
begin
  i:=l;
  j:=r;
  w:=random(r-l-1)+l;
  p:=a[w];
  k:=c[w];
  repeat
    while (((a[i]=p)and(c[i]<k))or(a[i]>p))and(i<r) do
      inc(i);
    while (((a[j]=p)and(c[j]>k))or(a[j]<p))and(j>l) do
      dec(j);
  if i<=j then
    begin
      q:=a[i];
      a[i]:=a[j];
      a[j]:=q;
      q:=b[i];
      b[i]:=b[j];
      b[j]:=q;
```

```

        q:=c[i];
        c[i]:=c[j];
        c[j]:=q;
        inc(i);
        dec(j);
    end;
until i>=j;
if i<r then qsort(i,r);
if l<j then qsort(l,j);
end;
begin
    init;
    readln(n,q,r);
    for i:=1 to 2*n do
        read(a[i]);
    readln;
    for i:=1 to 2*n do
        begin
            read(b[i]);
            c[i]:=i;
        end;
    qsort(1,2*n);
    for i:=1 to q do
        begin
            for j:=1 to n do
                if b[2*j-1]>b[2*j] then
                    inc(a[2*j-1])
                else
                    begin
                        inc(a[2*j]);
                        p:=a[j*2];
                        a[j*2]:=a[j*2-1];
                        a[j*2-1]:=p;
                        p:=b[j*2];
                        b[j*2]:=b[j*2-1];
                        b[j*2-1]:=p;
                        p:=c[j*2];
                        c[j*2]:=c[j*2-1];
                        c[j*2-1]:=p;
                    end;
            end;
        qsort(1,2*n);
        end;
    writeln(c[r]);
    clo;

```

模拟

end.

测试情况：

题号	时间(s)	题号	时间(s)
1	0.03	6	0.54
2	0.03	7	1.70
3	0.03	8	2.24
4	0.12	9	2.87
5	0.25	10	3.94

优化：注意到每轮结束后，赢的一方皆有序，输的一方皆有序，故可采用合并排序。

时间效率可降为 $O(n \log n)$

代码：

```
var
  a,a1,a2,b,b2,b1,c,c1,c2:array[0..200000] of longint;
  p,q,r,i,n,j,l1,l2:longint;
procedure init;
begin
  assign(input,'swiss.in');
  assign(output,'swiss.out');
  reset(input);
  rewrite(output);
end;

procedure clo;
begin
  close(input);
  close(output);
end;

procedure qsort(l,r:longint);
var i,j,p,q,k,w:longint;
begin
  i:=l;
  j:=r;
  w:=random(r-l-1)+l;
  p:=a[w];
  k:=c[w];
  repeat
    while (((a[i]=p)and(c[i]<k))or(a[i]>p))and(i<r) do
      inc(i);
    while (((a[j]=p)and(c[j]>k))or(a[j]<p))and(j>l) do
      dec(j);
    if i<=j then
```

```

begin
  q:=a[i];
  a[i]:=a[j];
  a[j]:=q;
  q:=b[i];
  b[i]:=b[j];
  b[j]:=q;
  q:=c[i];
  c[i]:=c[j];
  c[j]:=q;
  inc(i);
  dec(j);
end;
until i>=j;
if i<r then qsort(i,r);
if l<j then qsort(l,j);
end;

begin
  init;
  readln(n,q,r);
  for i:=1 to 2*n do
    read(a[i]);
  readln;
  for i:=1 to 2*n do
    begin
      read(b[i]);
      c[i]:=i;
    end;
  qsort(1,2*n);
  for i:=1 to q do
    begin
      for j:=1 to n do
        if b[2*j-1]>b[2*j] then
          begin
            inc(a[2*j-1]);
            a1[j]:=a[2*j-1];
            b1[j]:=b[2*j-1];
            c1[j]:=c[2*j-1];
            a2[j]:=a[2*j];
            b2[j]:=b[2*j];
            c2[j]:=c[2*j];
          end
        else

```

```

begin
  inc(a[2*j]);
  a2[j]:=a[2*j-1];
  b2[j]:=b[2*j-1];
  c2[j]:=c[2*j-1];
  a1[j]:=a[2*j];
  b1[j]:=b[2*j];
  c1[j]:=c[2*j];
end;
l1:=1;
l2:=1;
while (l1<=n)and(l2<=n) do
  begin
    if (a[l1]>a2[l2])or((a1[l1]=a2[l2])and(c1[l1]<c2[l2])) then
      begin
        a[l1+l2-1]:=a1[l1];
        b[l1+l2-1]:=b1[l1];
        c[l1+l2-1]:=c1[l1];
        inc(l1);
      end
    else
      begin
        a[l1+l2-1]:=a2[l2];
        b[l1+l2-1]:=b2[l2];
        c[l1+l2-1]:=c2[l2];
        inc(l2);
      end;
    end;
  while l1<=n do
    begin
      a[l1+l2-1]:=a1[l1];
      b[l1+l2-1]:=b1[l1];
      c[l1+l2-1]:=c1[l1];
      inc(l1);
    end;
  while l2<=n do
    begin
      a[l1+l2-1]:=a2[l2];
      b[l1+l2-1]:=b2[l2];
      c[l1+l2-1]:=c2[l2];
      inc(l2);
    end;
  end;
end;
writeln(c[r]);

```

合并排序




```

clo;
End.

```

测试情况：

题号	时间(s)	题号	时间(s)
1	0.03	6	0.17
2	0.03	7	0.29
3	0.03	8	0.39
4	0.06	9	0.54
5	0.09	10	0.63

4.算法类似于表达式计算，一个符号栈，两个数据栈。记 $f(s,0)$ 表示表达式 s 为 0 的方案数， $f(s,1)$ 表示表达式 s 为 1 的方案数。 $f(a+b,0) = f(a,0)*f(b,0)$ ， $f(a+b,1) = f(a,0)*f(b,0)+f(a,0)*f(b,1)+f(a,1)*f(b,0)$ ， $f(a*b,0)=f(a,0)*f(b,0) + f(a,1)*f(b,0) + f(a,0)*f(b,1)$ ， $f(a*b,1) = f(a,1) * f(b,1)$

注意：此题数据量大，非 $o(n)$ ， $o(n\log 2n)$ 算法均不易 AC.

代码：

```

const p=10007;
var x0,x1,y0,y1,z0,z1,n,i,j,n2,top1,top2:Longint;
    a,b,st2:array[0..200010] of char;
    st10,st11:array[0..200010] of longint;
    level:array['!..'Z'] of longint;
    ch:char;
    f:boolean;
procedure calc;
begin
    y0:=st10[top1]; y1:=st11[top1]; dec(top1);
    x0:=st10[top1]; x1:=st11[top1]; dec(top1);
    ch:=st2[top2]; dec(top2);
    if ch='+' then
    begin
        z0:=y0*x0 mod p;
        z1:=(y0*x1 mod p+x0*y1 mod p+x1*y1 mod p) mod p;
    end
    else
    begin
        z0:=(y0*x0 mod p+y0*x1 mod p+x0*y1 mod p) mod p;
        z1:=y1*x1 mod p;
    end;
    inc(top1);
    st10[top1]:=z0;
    st11[top1]:=z1;
end;
begin

```

```

assign(input,'exp.in'); reset(input);
assign(output,'exp.out'); rewrite(output);
readln(n);
for i:=1 to n do read(a[i]);
a[0]:='('; a[n+1]:=')';
n2:=0;
for i:=1 to n do
  if (a[i]='+') or (a[i]='*') then
    begin
      if a[i-1]<>')' then
        if b[n2]<>'X' then
          begin
            inc(n2); b[n2]:='X';
          end;
        inc(n2); b[n2]:=a[i];
      if a[i+1]<>'(' then
        begin
          inc(n2); b[n2]:='X';
        end;
      end
    end
  else
    begin
      inc(n2); b[n2]:=a[i];
    end;
n:=n2;
level[':']=0; level['+']=1; level['*']=2;
top1:=0; top2:=0;
for i:=1 to n do
  if b[i]='X' then
    begin
      inc(top1);
      st10[top1]:=1;
      st11[top1]:=1;
    end
  else if b[i]='(' then
    begin
      inc(top2);
      st2[top2]:='(';
    end
  else if b[i]=')' then
    begin
      while st2[top2]<>'(' do calc;
      dec(top2);
    end

```

```

else
begin
f:=true;
while f do
if top2=0 then
f:=false
else
if level[b[i]]<=level[st2[top2]] then
calc
else
f:=false;
inc(top2); st2[top2]:=b[i];
end;
while top2>0 do calc;
writeln(st10[1]);
close(input); close(output);
end.

```

测试情况：

题号	时间(s)	题号	时间(s)
1	0.03	6	0.04
2	0.03	7	0.03
3	0.03	8	0.03
4	0.05	9	0.04
5	0.04	10	0.04